

OZAmiga

Magazine

Volume 1 Edition 7

RRP \$6.95

News Flash
Guild Hall
Fantasy
Competition

ASSEMBLY
BASIC
AMOS
C

Make sure
you have your
coverdisk

BEGINNINGS
for the new user

V-Lab

YES



**Back Issues
are still
available**

***OZAmiga is produced
entirely on an A2000
using Pagestream 2.2***

<p><i>Pages are output to linotronic at the Typehouse in Perth</i></p>	<p><i>Printing is done by Mercury Press also in Perth</i></p>
---	--

***Distribution is handled by Newsagents
Direct Distributors***

OZAmiga Magazine

This months cover pic.

Mark Johnson has drawn this terrific Tools Table picture specifically for the coverage. He does most of his work using Imagine.

Regulars.

Letters	2
Programming in Assembly (Chris Leathley)	5
AMOS (Niel McKnight)	8
Learn C (Mark Little)	10
Beginnings (David Reeves)	16
CoverDisk	22
Over The Wire (Raff Lerro)	23
Brush Mapping (Mark Johnson)	26
Public Domain (Bill Holder)	28
Amiga Basic (David Perkovic)	32
Guild Hall	33
ArcAid	34

Specials.

Confessions of a Modem Addict	11
So you want to write Shareware (P Rosenboom)	12
V-Lab (Mark Scholmann)	18
The great Australian Rip Off (B. Holder)	20
Superbase 4 Pro (Neil McKnight)	30

Games:

Cryptic Crosswords	25
Super Frog	35
Lemmings 2	36

CONTENTS

THE CREW

David Reeves	EDITOR/ ADVERTISING MANAGER
Chris Leathley	PROGRAMMING EDITOR
Neil McKnight	AMOS EDITOR
Juan	COVERDISK ADVISOR
Raff Lerro	TECHNICAL CONSULTANT

Contributing Authors

Mark Johnson Bill Holder
David Perkovic Mark Little
Perry Rosenboom Bill Wheeler
Mark Scholmann

Disclaimer

The OZAmiga is an entirely independent publication striving to be a clear voice for the many talented Amiga users throughout Australia. The publishers do not accept any liability for any incorrect statements or information found in articles contributed by independent authors. The OZAmiga magazine in no way condones the illegal duplication of commercial software.

Subscribe and Save

YES !! I would like to subscribe to future copies of OZAmiga

Enclosed is a cheque/money order
for \$__ being for a __ edition
subscription.

Make cheques payable to,
Triple D Publications
PO Box 567
Mirrabooka WA 6061

Subscription Rates

3 Edition \$22
6 Edition \$35
12 Edition \$70

Subscriber Details

Name _____

Address _____

Suburb _____

State and Postcode _____



Here we are at edition seven and many thought we wouldn't make it. It is only through the continued support of our readers that we have kept the OZAmiga magazine going. I would like to thank all of those who have written in giving their views and criticisms as each is a form of contribution.

This edition sees a few minor changes, with Bill Holder taking over the PD section because the programs I was looking at were too old...hrumph...anyway it's sure to be a big improvement.

In the letters section you will find letters addressed to Raff Lerro, Chris Leathley, Neil McKnight as well as to myself. Some of the letters sent in were about technical things so they were immediately handed to Raff. We would like to see a lot more of this sort of thing as the

Question and Answer pages seem to help a greater percentage of people.

I have to also get down on my knees and beg forgiveness from all those people that had trouble with the coverdisk on edition six. It seems we neglected to look after our V1.3 users...Sorry! As I have said before, if you have any problems with your disk just send it back to us and we will rectify the problem if possible. If not, then we will replace the disk.

I have begun a much requested section for beginners, which starts at using the basic Workbench commands and will go on to show how to set up preferences and utilise some of those wonderful little PD utilities that everyone talks about.

I seem to ramble uncontrollably, so I will clam up and let you get on with the important stuff.

Till next one,
Dave

Dear David,

Congratulations to yourself and the crew for the production of a fine magazine dedicated to a great computer.

With reference to your "OZAmiga in the Future" section on page 37 of edition 5, I would like to express great interest in seeing some of the uses to which the Amiga is put within the business world. Also how about some information on business software and the use of Compugraphic and Adobe fonts.

I am presently using an A500 (Imeg), twin floppies and an Olivetti JP 150 Bubblejet printer for both business and home use. I am currently using this system mainly to write letters, produce training manuals, accounting, databases and DeskTop Publishing. I have previously used the Amiga in a Bookkeeping and Secretarial business which I happen to manage.

During a three month period this business was using the Amiga, an Apple Mac and an IBM 386. It was a very good opportunity to compare the three. The Apple was good to use but the printed output was dismal, while the IBM 386 was OK to output on but Windows is a pain in the neck when compared to Workbench. The 386 also suffered from heat exhaustion - the hotter the day, the slower it ran! (We were in a non-airconditioned office).

On an all round basis the Amiga come out on top despite lacking a hard drive and suitable software. Most of the DTP was done with Pagesetter V1.0 - it's amazing what you can produce with just a bit of ingenuity.

Despite the simplicity of my

setup (Oh how I long for an A1200/A2500 or an A3000), I have found it a flexible and easy to use system. It has also been greatly appreciated by my three children - I've even got my three year old using it now!

Best wishes for the future of OZAmiga, please feel free to call me if you need my help.

Yours sincerely,
Phil Johnson
Subiaco WA

Dear Phil,

There is not a great deal to say in answer to your letter, it was more of an information exchange than a query. Never the less, I would like to thank you for the comments on the magazine and your input towards it.

I am currently trying to sort out a few font troubles myself, including the use of Compugraphic, Adobe and Postscript fonts. It seems that the PD postscript fonts I was able to purchase, work fine when output from my Amiga to a HP DeskJet printer but when I try to actually output to postscript, I run into all sorts of problems. Anyway, I have the help of people in the typesetting business so I should be able to sort it out.

When I have finally resolved my little troubles I will undertake to put my results into an article (or series of articles, depending on how complex it gets) to help others overcome some of the same types of problems.

The proposed Amigas at Work section is due to begin in edition eight, so keep an eye out for that.

Regards,
Dave

Dear Raff,

Thank you for the brief but informative report on the A1200. I would have to agree with you that this is truly the future of Amiga computing.

However, I have a few unanswered questions.

Is the A1200HD IDE controller really an A2091 controller? According to AmaxII+ it is. For some unknown reason AmaxII+ will not boot up from my Amax partition on my hard disk. In fact the system hangs whenever it tries to. PLEASE HELP A DESPERATE AMAX USER (because Readysoft won't).

Why are those two keys (the one on the left of the 'Z' key and the other one which is below the '}' key) not labelled?

Forgive me for being stupid but can I not get a technician to solder a 68882 onto the A1200's motherboard?

Why is Commodore still supplying such a crappy mouse with the A1200?

Why do the graphics look so jaggy on a 1960 and not on a 1084?

How can the Amiga community effectively stop the IBM and MAC propaganda? It has been an uphill battle for me!

For those of you who have troubles running games on the A1200 because of WB 3.0, there is a program called RELOKICK 1.3 which brings backwards compatibility to most WB2.0

& WB3.0 machines (yippee). You will find it on disk 52 of CU Amiga magazine (I also buy OZAmiga too, of course).

Thank you for your time and keep the mag going.

George Lui
Ascot Vale VIC

P.S. Here are some internet addresses for Amiga downloading:

128.193.124.2 (US)
128.252.135.4 (US)
130.130.68.4 (Australia)
192.107.107.6 (Australia)
131.234.2.32 (Germany)
131.188.1.47 (Germany)

Do I win editions 1,2 of OZAmiga for doing that?

Dear George,

After looking at the A1200, I would say that it's IDE interface is not like the A2091 much at all.

The A2091 controller has a SCSI interface as well as IDE, but I think the IDE side is XT IDE, not AT IDE as in the case of the A1200. The difference, although it may seem small, is significant, an AT IDE interface is designed with an IBM AT type computer in mind and is thus 16 bits wide, an XT IDE is designed with an IBM XT style computer in mind, and is thus only 8 bits wide.

In the case of an A1200, the controller built into the motherboard would appear to be an AT style and is thus 16 bits wide, unlike the A590/A2091 XT IDE controller. This may be the cause of Amax's confusion, but although IDE controllers are meant to be a standard of some sort, there are sometimes compatibility problems with different drives etc. I know this probably doesn't help you a great deal, but I can think of no easy cure to your dilemma. Let's hope that Readysoft produce a revamped version of Amax, if that is the problem, or maybe a patch to the program.

As for the unlabelled keys found on the A1200, I'm told that they are for different language versions of the keyboard, part of WB2.1 and WB3.0 locale additions I'd suspect. On Australian keyboards I don't think the extra keys do anything.

As for the addition of a Maths chip (68882) to an A1200, I would imagine that the pads on the motherboard are there for some future model upgrade, or just a change of specification in the final stages of the A1200's design, with the Co-Processor left out.

Theoretically, it would be possible to

solder a chip onto these pads, but as the chip would be surface mounted (read very small and with no legs as such), it would be rather a difficult job without special tools. You may be able to find a technician to try to solder a chip in for you though, but as I don't have an A1200 to experiment upon, I can't say for sure that it would work, or if you would need additional hard or software to enable the co-processor.

As for Commodore supplying the "crappy" mouse, well I can't really answer that one, perhaps it is a matter of economics, or styling, who knows. Commodore themselves would best be able to answer this question.

The resolution available using a 1600 monitor is somewhat better than that of the horrible 1084, especially if you take advantage of the higher resolution modes available when using a multi sync like the 1600. Because of this I'd say that's why the graphics may look jaggier on a 1600. Were you using the same screen modes on both monitors?

How best to stop the propaganda? Well, if you ask me, competition is good, and the best way to help the Amiga along is to demonstrate its stunning performance and potential to everyone who'll look at it. True some of the hype that gets about is somewhat less than the complete truth, but there isn't a lot we can do about it. Just show your Amiga off to IBM and MAC heads every chance you get, I do!

You'll have to ask the Editor about some back issues, as it isn't me who makes those kind of decisions. Good luck!

Regards,
Raff.

Dear Raff,

My Amiga has suffered a most unworthy punishment. I have a 1Meg, 1.3 KickStart A500, with 1 external drive (this is just a bit of a background that might help solving my problem). A friend sent me some text/pic files that document how to make a device similar to the X-Copy cyclone hardware device. I thought great, now I can back up my software that I never could. Well, I finally found a program that could work in conjunction with the device. But it required 2 drives, no problem, except with the hardware device plugged in the computer didn't recognise the external drive.

So I ran the program without the device, and unplugged the drive and put the hardware device between the drive and the computer. Again no problems. But when I next decided to run WorkBench it decided not to work. WorkBench runs okay, but I can only click on the disk icon once and after that nothing happens.

If I hit the right mouse button the computer hangs. This also happens with a few other programs, but WorkBench is the worst. I decided to leave the computer in its hung state and after 5 or so minutes it seemed to catch up to itself and open the disk, etc. But this only happened once.

I realise I must have partially blown one (or more) of my control chips and I'm sure its either the Gary or one of the CIA chips as the timing appears bugged. I (finally) am asking if you happen to know whether I'm correct or has something else gone wrong as I don't really have the money to just buy a Gary chip and several CIA's and check.

Thanks for your time, and keep up with the great mag only I wish it were a little bit bigger, and are back issues available?
Rodney Norton.
Leumeah, NSW.

Dear Rodney,

Thanks for the comments on the mag, as it is always good to receive feedback. I'm sure as time progresses, that the magazine shall continue to grow, and I know that it appears a little thin compared to some others, but as OzAmiga contains little advertising, you still get good value for money I think. As for the back issues, I think that most are still available, but the Editor may like to comment more on this.

Now for your problem.

I too have built a Hardware device similar to the one for Cyclone, and have these observations that may help you.

When the device is connected in series with your external drive, the WorkBench, and nearly all other AmigaDos type programs will be unable to address the external drive correctly. In fact, on my machine with the device connected the WorkBench also doesn't see that df1: is connected. And if using V2.05 KickStart, the WorkBench has df1:??? displayed.

I have found that this is normal, as the device is only meant to be used in conjunction with a program such as Cyclone, and not designed to be left connected to run WorkBench etc.

The reason for this is that in the design of the device, the designer used some very tricky hardware kludging to change the way that the drive interfaces with the Amiga, and the associated copying program then uses this to copy some of the more heavily protected software around.

So if your Amiga works fine without the device attached, and the device works with the associated program that it was designed for, then you no longer need worry. But if your Amiga is still faulty without the device attached, then you

may well have destroyed a CIA chip, especially if you were disconnecting and reconnecting drives with the power turned on, in fact to plug or unplug ANY peripheral into the Amiga with the power on, is asking for trouble, as it is quite easy to damage the internals of the Amiga this way.

Looking at my circuit diagram for the A500, the CIA marked U8 would be my best guess as being the faulty one, but you could try swapping the two CIA's and trying the machine again, to see if the fault has changed to being on the serial or parallel port, then you may be able to tell which CIA is faulty, as U8 also controls the serial port hand shaking, and the other CIA, U7, controls most of the parallel port in addition to the keyboard etc.

But unfortunately, that may not be the problem, as the Gary chip may also be faulty, and really can only be checked by substitution, and there may also be a faulty TTL chip, especially if the drive was unplugged with the power on.

I hope these suggestions help, and good luck!

Regards,
Raff.

Dear OZAmiga,

Congratulations on what must be the first ever decent "learn to program" section in any magazine. Chris Leathley's assembly column is exactly what I've been needing for quite some time. I hope that his articles will continue to be as well written and informative as his article on Copper.

When I first bought my A500, my main aim was to learn assembly programming, just as I had on the C64. It is now six years later and I still haven't learnt it!! The problem being that I haven't had the time or the money to be able to get into it 100% so I keep putting it off.

Just recently Amiga Format featured a programming column by Bullfrog but unfortunately it was pretty disappointing. They basically just worked on controlling a program without going into the really important stuff like the 68000 control or the Copper. It's no good being able to write a fantastic inertia routine (etc) when you have no idea how to set up a screen and display a sprite on it.

Then along comes OZAmiga to the rescue. Parts one and three of Chris's column were regular stuff (I never got Ed two), but the section on Copper was fantastic! Luckily, I now own Devpac 2 from the aforementioned Amiga Format

thingy, so I was able to have a good look at Chris's examples. Working through the lessons, I realise that the Amiga is not that much different to the old C64 (in terms of how it works), so I guess I'm already half way there. What I would like to see though, is a lot more information per issue. As much as the Copper article was very informative (and well written), if you only cover a couple of features each edition, you will still be going into next century!

As far as future articles go, I would like to see a *complete* run down on the Copper, an extensive look at setting up screens (the Copper demos did have some good example code!), information on displaying pictures and something on the Blitter. If Chris can give some info on any of that and in the same clear form that previous articles were written in, then it would be extremely useful to me at least.

Could you also be sure to make each editions work compatible with previous routines. For example, if, in the future Chris describes the process of displaying a logo/picture on the screen, then enough information should be given to allow the reader to use the Copper effect on the same screen.

A couple of months back you did a story about Chris's game FORTRESS. Being a big fan of the old C64 classic, FORT APOCALYPSE, I'm keen to hear how it's going. An update might be good.

Also do you know anybody in Australia who produces commercial games for a living (does Chris?), because I'm interested in acquiring a job as an artist. I'm just about to send eight packages to various computer companies in England with the hope of finding work (as either a C64 programmer or as an artist for Amiga games) and just thought that if there were any programmers in Australia looking for someone to produce graphics (full-time only), then it might be worth trying them as well.

Rowan Crawford
Victoria

Dear Rowan,

This letter was passed directly to Chris Leathley as he was best able to answer your queries.

Thanks for the praise indeed. I'm glad you find the tutorials informative and well written. I do try and do my best. Like you mentioned I do go a fair bit into the technical stuff while at the same time trying to keep it at an English level. The good old 64 and the Amiga are quite similar but as you know the 64 uses character while the Amiga uses bitmap. Look in this issue's tutorial for a run

down on bitmaps.

As for Fortress, maybe we could do an other "work in progress" as it has come a long way since late last year. I don't work as a games programmer full time only part time as Australia doesn't have much to offer programmers like me.

I have to work under contract through English software houses. Many people like me are developing games so you maybe able to find some work for your artistic skills, so keep trying.

Hope you do well and keep on hacking at 68000.

Chris...

Dear David,

I'm a new reader of your magazine, OZAmiga, and am glad to see another Australian Amiga magazine. I was particularly interested in vol1 ed6 because of the article on **Virtuality** systems and their locations. You will be pleased to know that they are not all "Travelling" as stated in your article. I managed to track down one in North Ryde RSL (in NSW) and it is now permanently installed there at a cheaper rate than the system in Perth.

Apparently since May the club has been holding VR tournaments on Sunday afternoons for only \$2 entry fee in the **Splinters** bar at the RSL.

There is a prize of \$100 offered each week and I won after only a few games practice.

Here are the clubs details, if you or your readers would like to contact them for more information:

North Ryde RSL Community Club
Magdala Road North
North Ryde
NSW
Phone: (02) 888 7588

I hope this information helps those in search of a Virtual experience.

Sincerely
Andrew Trickett
NSW

Dear Andrew,

I thank you very much for this information as it is a bit difficult to pinpoint some of these machines sometimes. With many people all over the country itching to get a go on a Virtuality machine, you have definitely filled a need with your letter.

Thanks again,
Dave

BITPLANES, PLAYFIELDS and BITMAPS

Welcome back guys, does everybody understand copperlists now? I've had a few people mailing me compliments so everything must be going ok.

In this installment (yes people I'm writing a book (It seems like that anyway (he he))) I will be covering Amiga bitplanes. This subject covers quite a lot of ground and can be quite technical, because of this I will try to explain it all as simple as possible. Wish me luck people! (FOOTNOTE: if I do fail then at least the example programs will help)

Bitplanes, Playfields, or Bitmaps as people call them make up the basic display element of the Amiga. The copperlist controls DMA output to the monitor but apart from changing pointers and colours doesn't put much visible on the screen. The picture you look at when you first turn on amiga or the most spectacular game graphics are all formed in bitmaps in the Amigas CHIP memory.

Playfields are constructed from 1 to 6 bitplanes or bitmaps, they allow the Amiga to have a range of display outputs:

- 2 - 4096 colours on the screen at once.
- A picture resolution of 16 by 1 to 704 by 625 pixels.
- Two independent playfields are supported
- Smooth scrolling at all directions (hardware assisted)

Playfields can be made up of most combinations of the above to create displays from the 640 by 256 workbench screen to the latest mind bending video game with zillions of colours on screen. (not quite but that's what they want you to believe)

BITPLANES

The Amiga has four basic display modes - low resolution, high resolution, interlaced and non-interlaced. In a low resolution picture there are 320 pixels across while in high resolution there are 640 pixels in the same physical display area. The same with interlace (400 pixels down) and non-interlace (200 pixels). These values are normal dimensions, they can be made smaller or larger which is called "overscan". Even though these modes share the same physical size they do differ in the amount of memory required, this must be considered when choosing a display mode.

A pixel is a sequence of binary 1's and 0's that is used to determine the colour

to use. The more bitplanes, the more colours can be used. As mentioned above in a round about way there can be upto 6 bitplanes that form a picture. The 6'th bitplane is used for several special purposes and will be explained at a later date.

There are 32 separate colour registers in the hardware address area and these are used by combinations of bits from the first 5 bitplanes. The number of bitplanes is logarithmic to the number of colours available.

ie. 1 bitplane can have 2 colours. Pixel on (colour 1), pixel off (colour 0). 2 bitplanes can have upto 4 colours, 3 bitplanes = 8 colours, 4 bitplanes = 16 colours and 5 bitplanes has the full 32 colours.

Imagine this diagram to be 3 dimensional.

A 1 Bitplane picture

0 = Colour 0
1 = Colour 1

A 2 Bitplane picture

0 & 0 = Colour 0
1 & 0 = Colour 1
0 & 1 = Colour 2
1 & 1 = Colour 3

FORMING A PLAYFIELD

To get started we must understand what a bitplane is and how it is stored in memory. The normal display size for a bitplane is 320 by 256 pixels, since these are really binary values then each byte can hold 8 pixels. 320 pixels divided by 8 (a byte) results in a count of 40 bytes. Times that by the height (256) and you get 10240 bytes or 10K. If you have 4 bitplanes then you would require 40960 or 40K (If the playfield size is smaller or larger then the amount of memory needed changes). Each bitplane is a separate chunk of memory and can be anywhere in CHIP type memory. There is constraint with the width of a bitplane, it must be divisible by 16 or must have

an even number of bytes. So a bitplane of 322 pixels is not possible, so 336 is the nearest value that we can use. Unlike binary as we know it (a stream of 0 and 1 bits) the display hardware gets each of it's display pixels from different parts of memory (so the next binary digit of a colour can be any amount of distance away instead of the next bit). The display hardware overlaps these bitplanes to form the display and must be told where each bitplane starts in memory.

The hardware has a group of bitplane address registers stored at \$00DFF0E0 to \$00DFF0F4. These are long word addresses and can be written as such, ie. move.l #\$00020000,\$00DFF0E0 would put the address of our bitplane (\$00020000) into the display hardware.

The complete address list is:

\$00DFF0E0 - Bitplane 1 Address
\$00DFF0E4 - Bitplane 2 Address
\$00DFF0E8 - Bitplane 3 Address
\$00DFF0EC - Bitplane 4 Address
\$00DFF0F0 - Bitplane 5 Address
\$00DFF0F4 - Bitplane 6 Address

Unfortunately we have to write these registers with the copper to stop DMA clashes. As we all know the copper can only write 16 bits or a WORD into memory so we must split each bitplane address up again into two parts. The top 16 bits of a longword \$00DFF0E0, and the bottom 16 bits \$00DFF0E2. The example program BITPLANE1.s shows how the copperlist is configured and how to write the address of our bitplane into these registers and into the copperlist.

BITPLANE CONTROL

\$00DFF100 or BPLCON0 is the bitplane control register. This tells the system what mode to use and how many bitplanes to display aswell as interfacing other video equipment.

BIT No.	NAME	FUNCTION
15	HIRES	High resolution mode on (HIRES = 1)
14	BPU2	These three BPUx bits tell us how many bitplanes to display.
13	BPU1	
12	BPU0	
11	HOMOD	Hold and modify mode on (HOMOD = 1)
10	DBPLF	Dual playfield on (DBPLF = 1)
9	COLOUR	Video output colour (COLOUR = 1)
8	GAUD	Genlock audio on (GAUD = 1)
7-4	-	Unused
3	LPEN	Light pen input active (LPEN = 1)
2	LACE	Interlace mode on (LACE = 1)
1	ERSY	External synchronization on (ERSY = 1)
0	-	Unused

Most of these options above won't be needed but must be told about to help explain any missing blanks in registers. With the new AGA (Amiga 1200 & 4000) CHIP set more hardware registers have been added and some of the previously unused bits changed to be active. It is best not to mess with these as what works on one machine may not work on another.

This register works like a normal memory address, so it doesn't modify single bits, the whole word must be written at once. We will be using bits 9 and 12-14. If we want to turn on a single bitplane we would move \$1200 into \$00DFF100 or if we wanted 4 bitplanes then \$4200 would be needed. The reason we turn on the video colour output is that if anybody is using some weird monitor configuration then they will receive a colour signal instead of a black and white one.

There are two other Bitplane control registers (\$00DFF102 and \$00DFF104) that control the hardware scroll and bitplane priority but these will be explained next issue and will be set to \$0000 for now.

DEFINING THE SIZE OF THE DISPLAY WINDOW

Now this is where things get hairy. When you have decided what type and the size of playfield you must tell the hardware the size of the display window, which is the actual size of the on-screen display. The adjustment of the display window affects the entire display mechanism including the border and sprites, not just the playfield. You can't display anything outside this window.

The size of the playfield is not directly related to the window size. The window size can be smaller allowing smooth scrolling or to display only a portion of the entire playfield.

You define the size of the window by specifying the vertical and horizontal position at which the window starts and

stops. The resolution of the vertical start and stop positions is 1 scan line. This works exactly like the copper wait command, allowing you to time the display with the copper output. The horizontal start and stop position resolution is 1 pixel. These positions are given in an (y,x) order. The window can start at (0,0) but as FIGURE 1 shows (0,0) is way out of the visible screen boundaries (it's sitting in space 1 inch outside of the monitor box). The usual starting position for a 320x256 picture is \$2C for the vertical and \$81 for the horizontal. The hardware allows you to specify a starting position before this value but not all of it may be visible. The difference between the absolute starting position (0,0) and the normal position (\$2C,\$81) is the result of the way many video monitors are designed. To overcome the distortion that can occur at the extreme edges of the screen, the scanning beam sweeps over a larger area than the front face of the screen can display. A starting position of (\$2C,\$81) should be central, leaving a border of about 8 pixels around the display area. On TV's this is not always true as the raster beams are more rigid to display PAL TV pictures so the display is often offset to the right. The stopping position for the above size display is (\$2C,\$C1).

bitplane DMA is turned off. This is important as you will mostly likely get DMA clashes. A good example of where this is used is when you pull down

a screen that overlaps another. Between the screen boundaries there will be 3 blank lines where the bitplane isn't displayed and if you move your mouse over it, some lines will disappear within the sprite graphic. That's because they have to turn off bitplane DMA to change bitplane pointers, window sizes, resolution etc.. An odd side effect is that sprites cannot be displayed in this area as they require bitplanes to run! but we get into that when we do sprites. We only need to turn off DMA for 1 whole raster line, not 3 like workbench.

To define our window sizes we write the WORD \$2C81 into \$00DFF08E and \$2CC1 into \$00DFF090. You may notice that the vertical start and stopping positions are the same, because we can only specify bytes the vertical positions have been put into boundaries. The start position ranges from \$00 to \$FF which is 2 inches from the bottom of the screen while the stop ranges from \$39 (top half of screen) to \$138 (bottom half). We only write the byte value of \$38, the hardware assumes you mean the bottom of the screen. This means you cannot have a very small window at top or bottom of the screen using the window size variables only, we can get around this by turning off the bitplane DMA at the required point. The window sizes are usually only specified once.

DDFSTR and DDFSTOP

After defining the window size and position, you need to tell the system the on-screen location for data fetched from memory to start and stop. To do this, you give it the horizontal start and stop positions. The data fetch registers have a four pixel resolution (unlike the display window registers). It is recommended by

Commodore that the data fetch start value be restricted to a resolution of 16 pixels or a count of 4.

The normal low-resolution DDFSTR (\$00DFF092) is \$0038. We can make

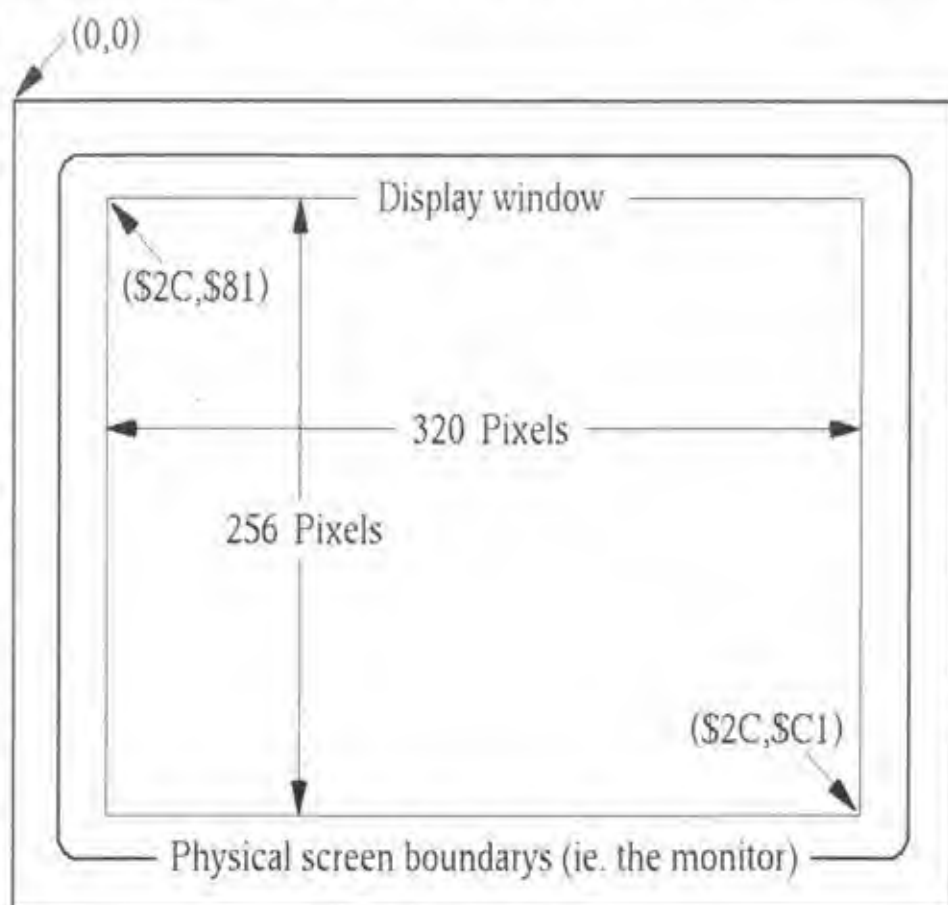


Figure 1

We have two hardware registers called DIWSTRT (\$00DFF08E) and DIWSTOP (\$00DFF090) these are written by the copper either in the vertical blanking gap or when no

this smaller so it starts getting data from memory earlier. Programmers do this to allow them to get smooth scrolling in games, as the computer does a bit of the work for them (not quite all though (he he)). There is a simple maths equation that allows us to calculate the DFFSTRT from our window size.

$$((\$81 / 2) - 8) = \$2C$$

The relation ship between data-fetch start and stop is:

$$\text{Width in pixels} - (\text{DDFSTRT} * 2) \\ 320 - \$70 = \$D0$$

so writing \$00D0 into DDFSTOP (\$00DFF094) will tell the dma when to stop fetching data from memory.

MODULO

If we have a larger bitmap that what we are displaying on screen we need to tell the system the difference between the width of the on-screen window and the distance to the next line of our bitmap. For the normal size bitmaps we will be using a modulo of \$0000 as there is now difference. (ie we are displaying a bitmap of 320 pixels width on a screen 320 pixels wide). When we get into scrolling next issue all will become clear. For now we move \$0000 into both BPL1MOD (\$00DFF108) and BPL2MOD (\$00DFF10A).

DMA CONTROL

Now we have defined the window size, position and data-fetch values we must tell the system to turn on the bitplane DMA. Remember the DMA control register DMACONW (\$00DFF096) from last issue?

BIT	NAME	FUNCTION (when set)
15	SET/CLR	Set or clear bits
14	BBUSY	Blitter busy (read only)
13	BZERO	Result of blitter operations is 0 (read only)
12		Unused
11		Unused
10	BLTPRI	Blitter has priority over processor
9	DMAEN	Enable DMA set below (main switch)
8	BPLEN	Enable bitplane DMA
7	COPEN	Enable Copper DMA
6	BLTEN	Enable Blitter DMA
5	SPREN	Enable sprite DMA
4	DSKEN	Enable disk DMA
3-0	AUDxEN	Enable audio DMA for sound channel x

We must move #8380 into \$00DFF096 to instruct the hardware we want bitplane DMA. We also turn on copper DMA and enable.

BITPLANE COLOURS

As discussed in copperlists the address of colour 0 is \$00DFF180. All of the 32 colour registers follow directly

afterwards from \$00DFF182 - \$00DFF1BE. They move up in multiples of 2 bytes so colour 1 = \$00DFF182 while colour 2 = \$00DFF184. When you set up bitplanes you must also specify the colours to be used otherwise everything will be black. The copper is useful for loading the colour RGB values into these registers. The example programs will illustrate this.

CONVERTING DPAINT PICTURES TO BITMAPS

Also supplied on the disk is a program called IFFMASTER by Arcane. There is a better one called DELUXEIFF but it seems to have some problems on Kickstart 2. This program allows the user to load if dpaint iff pictures and save either the whole picture or parts of it as a raw bitmap. You can also save the colour palette information to be included in the copperlist.

LOADING A IFF PICTURE: Click on the RMB and activate "load picture" from the menu bars. The file requester will pop up allowing you to select what picture you want to load. (Try 4bp.oz in the assembler directory). The top of the screen will display information relating to the size of the bitmap, the number of bitplanes and the brush sizes.

SAVING WHOLE PICTURE AS A RAW BITMAP: Click on "save picture as raw data". The file requester will again pop up. (The most annoying thing about this program is that the file requester forgets where we were last). Make sure that you don't go over the IFF picture filename. The usual convention

it. This allows you to draw a box around the area you want to save as a smaller bitmap. The example IFF 4BP.oz (now you know my favorite band anyway) is only 112 pixels high and not the usual 256. Click on "brush" in the menu or press "B" on the keyboard. Move the mouse cursor around to the X value equals 0 and Y value is 23 (the cross hair line should overlap the top line of the text). This means you are at the top of our bitmap. Press and hold down the right mouse button, this will fix the top corner of the box at that position. Move the cursor down to and to the right until the BW (brush width) and BH (brush height) values are 320 and 112. You will notice that the box makes a vertical grid every 16 pixels that is because bitmaps must be a multiple of 16. Let go of the mouse button and our brush outline box should be complete. Click on "save brush as raw data" and save it as a ".raw" file.

SAVING THE COLOUR PALETTE: When we have complex, multiple bitplane picture we will also want to save the colour palette information. The program can generate to palette information in a copperlist format. To do this we must change the preferences of the program. Click on "preferences" in the menu and a list of options will come up. Click on "separate palette save" until "ASM source for copper-list" come up. Click on the exit window box or press "ESC". Now click on "Generate palette" and give the requester the output file name (append ".pal" to it). You can then use you text editor to copy the outputted source code into your own copperlist.

A bit of experimenting will enlighten you to the uses of the program. The other preferences will be explained at a later date as we get further into bitmaps.

On the Coverdisk in the Assembler directory there are 6 example programs using the how to use bitmaps with the copper. They also show multiple bitmaps and a vertical scroll of one of the bitmaps. Got slowly though these programs and understand, as usual they are well commented. Try and create your own programs using your own picture, after all as the saying goes, you must get you fingers dirty if you want any work done. If anybody has any difficulty then contact me at OZAmiga.

Next month we will be covering high resolution mode, dual playfields and simple 8 way scrolling using the amiga's hardware.

Till then live long and prosper.

ZAPHOD

SAVING PART OF A PICTURE: If you move the mouse pointer around then you will notice the cross hair following

In this issue we will deal with a few final graphics commands, for a while at least, and begin on the next section: Text & Fonts.

If you have been following the last few issues, we have already covered colours, drawing commands, screens and mouse control. Included in the last issue was the source code for a small paint program written using the commands learned thus far. If you have any problem with these or any other AMOS commands, please feel free to drop me a line at the address at the end of this article.

Colour Effects

In a previous issue we looked at colours and colour mixing. In this issue we will find out how to do fades and colour cycling effects. The first command we will look at is the Fade command. It can be used to smoothly fade all colours to black. The syntax for fade is:

Fade <n>

The value for <n> is the speed of the fade. This value ranges from 1 (very fast) to about 10 as a useful limit. The value <n> is the number of vertical blanks between each change in colour. A vertical blank occurs every 50th of a second. As there are 16 possible values for each component of a colour (see Graphics Part 1), a fade will take $16 * <n>$ to complete. It's a good idea to WAIT for that amount of time in your program before continuing.

Examples:

Fade 1: Wait 15 - Very fast fade.
Fade 2: Wait 30 - Nice quick fade.
Fade 3: Wait 451 second fade out.

Fade also has the ability to fade from black to a specific set of colours. Used this way, a Fade In effect is possible. The syntax for this version of the command is much like the Palette command covered in an earlier issue.

Fade <n>,<SRGB>,<SRGB>,<SRGB>

This again uses <n> to set the fade speed. The <SRGB> values are the colour values that each pen will fade to. The leftmost value is for colour zero, the next is colour 1 etc.

If you leave out a value, that colour will not be affected.

With:

Neil McKnight

Examples:

Fade 1,<FFFF>,<FFFF>,<FFFF>: Wait 15
fade colours 1-3 to white
very quickly.

Fade 1,<SCCC>,<S009>,<S066>: Wait 15
fade colours 1-3 to three
shades of grey.

If the two commands above were executed one after another, anything drawn in colours 1, 2, or 3 would flash white then fade to shades of grey. This effect can be used to create some attractive intro screens with little more than a picture and a few fade commands.

Colour Cycling

Anyone who has played with a paint program such as Deluxe Paint may have seen an effect known as colour cycling. Colour cycling causes an illusion of movement by rotating the colours of a set of pens in a loop, shifting the colour of each pen to the pen next to it. Pen 1 is copied to pen 2, pen 2 to pen 3 and so on. If you draw a series of lines on the screen using different pens, colour cycling makes the lines appear to move. There are three commands that control colour cycling. These are Shift Up, Shift Down and Shift Off.

Shift Up will rotate a series of colour registers moving the values from low to high. It is used like this:

Shift Up <d>,<s>,<e>,<t>

The value for <s> is the first pen you want to include in the cycle, <e> is the last pen. The variable <d> controls the speed of the cycle by waiting <d> 50ths of a second (vertical blanks or VBLs) between each step. The variable <t> is a control flag and can be either 0 or 1. If 1, then the value of the last pen will be copied to the first. If set to 0, then the value for the first pen will gradually fill the entire set.

To rotate colours in the opposite direction, use this command:

Shift Down <d>,<s>,<e>,<t>

As you can see, the syntax is the same as above.

To stop the colour cycling effect, use the command:

Shift Off

Something to note is that you can only have one Shift effect happening at a time. Also, colour cycling is controlled by an interrupt. This means that AMOS will take care of it while your program does other things. You only need to start it when you want it, and tell AMOS

when to stop. Closing a screen will automatically stop colour cycling, but clearing the screen won't.

Examples:

Shift Up 1,1,5,1 - fast colour cycle using colours 1 through 5.
Shift Down 1,1,5,1 - the same but in reverse direction.
Shift Up 25,24,31,1 - slow (2 per second) cycle using colours 24 to 31.

Image Compression

AMOS provides you with a means of storing screens in a compressed form. This allows you to draw a title screen for a game, for example, compress it to a fraction of its original size, and save it along with the rest of your program. When the program is run, your title screen can be decompressed and displayed in less than a second.

For programs that use a control panel of some sort, such as the Minipaint program in the last issue, this feature is invaluable. The required graphics can be compressed and stored inside one of AMOS's memory banks and displayed at will. Compressing images saves disk space and makes programming easier because you don't have to write loading routines or cope with missing files.

The simplest of these commands is Spack, which stands for Screen Pack. It is used like this:

Spack <s> To <n>

This will compress and store screen number <s> into bank number <n>. A Spacked screen contains all the necessary information to redisplay the original image exactly as it was. This includes the screen position, the palette, screen offset etc.

To display a compressed screen use the command Unpack.

Unpack <n> To <s>

This will decompress a Spacked screen from bank <n> to screen number <s>. If the screen was not already opened, it will open a suitable screen for you. It is also possible to Spack just a section of a screen.

Use this variation:

**Spack <s> To
<n>,<x1>,<y1>,<x2>,<y2>**

Here the area bounded by <x1>,<y1> and <x2>,<y2> forms a rectangle of the screen <s> that will be compressed into bank <n>. When Unpacked, the

resulting screen will be created the same size as the defined area. Note that all Spack commands will round the horizontal size to the nearest multiple of eight pixels.

If you want to display a compressed logo onto an already created screen, then the Pack command will be useful. Pack will compress and store just a section of the screen, but with the purpose of unpacking the graphics onto an already existing screen, rather than creating a whole new screen as Spack does. The syntax for Pack is:

Pack <s> To <n>

This works the same as Spack, but it expects a suitable screen to already exist to display the image on. Note that Pack does not store any colour information for the compressed image. The unpacked image will use the colours of the screen it is displayed on.

A section of a screen may also be stored into a bank by using this variation:

Pack <s> to <n>, <x1>,<y1>,<x2>,<y2>

As you can see, this is identical to the format for Spack.

To decompress your piece of graphics, use Unpack as before. When using Packed images, you may also use these variations of Unpack.

Unpack <n> - unpacks the image at it's original position.
Unpack <n>,<x>,<y> - unpacks the image at <x>,<y>

Graphical Text

There are two types of text printing in AMOS. Regular text uses characters that are always the same size and are displayed as rows and columns. Graphical text allows you to load a variety of fonts from disk, change type styles and print at any position on the screen you wish.

Using graphical text is easy. Use the Text command to display any string at any screen position.

Text <x>,<y>,<t\$>

This will display the string <t\$> at the screen coordinates <x>,<y> using the currently loaded font and style. The colour of graphical text is determined by the current Ink colour, the same as for drawing all other graphics commands.

Examples:

Text 10,10,"Hello" - top left corner.
Text 200,128,"Hi" - near the middle of the screen.
Text 100,100,3 - attr=3 To

display numbers, convert them into a string first, using the command Str\$().

Text 100,100,Str\$(3) - now this works.

Text Styles

You can change the style of the text, making it bold, underlined or italicised, or any combination of the three. The command for this is Set Text:

Set Text <s>

The value for <s> determines the style of the text according to the table below:

<s>	Bit 0	Bit 1	Bit 2	Style
0	0	0	0	Normal (plain)
1	1	0	0	Underline
2	0	1	0	Bold
3	1	1	0	Bold + Underline
4	0	0	1	Italic
5	1	0	1	Italic + Und/line
6	0	1	1	Italic + Bold
7	1	1	1	Italic + Bold + Underline

As you can see, there are eight possible styles that can be set. Another useful command will return the current value for style:

<s> = Text Style

If drawing your own buttons when using graphical text, it is useful to know how long the text will be before you print it, so you can work out the correct position. This command will help you:

<n> = Text Length(\$)

Also, you may need to know the height of the current font to make adjustments to th printing position. The command Text Base will give you the height of a font from the top of the tallest letter to the base line, which is where the letters "sit". Letters with descenders (gpq etc.) have parts that protrude below the base line, and this amount varies according to each font, so you will need to cope with this yourself.

Text Base is used as:

** = Text Base**

Loading Other Fonts

So far, all we have used is the normal AMOS font. With the following commands, we can load variety of fonts from disk. Disk fonts are found in the FONTS: directory of your Workbench disk.

Before we can load a disk font, we must first find out what fonts are available. The command to do this is Get Fonts.

Get Fonts - get all fonts.
Get Rom Fonts - get only the building ROM fonts (topaz).
Get Disk Fonts - get all fonts in the fonts directory

This command has a number of options, depending on which fonts you want to use. Generally, you want a Disc font.

Get Fonts by itself searches for the available fonts and builds a list of them internally. You will need to display that list so you can see the available fonts and select one to use. For this we do the following:

```
n=0
Repeat
  IS = Font$(n)
  Print Font$(n);IS";(n)
  Inc n
Until IS=""
```

This will display details about Font number <n> as stored in the internal list generated by Get Fonts, and help us to choose from among them.

To actually load the required font from disk, we use the command Set Font, and ask for the number as printed in the list provided above.

Set Font <n>

This will load the font from disk. Now, every time you use the Text command to print, your text will appear in that font.

To help you use disk based fonts in your programs, I have included some procedures that allow you to ask for a particular font and size by name. The procedures cope with the problem caused by the fonts being listing differently on different computers. For example, just because Helvetica 24 point is font #3 on your computer, doesn't mean it will be on mine. So you cannot just always ask for font #3, you need to find IF that font exists and what number it is if it does.

Next time, we will continue with text and text processing commands, have a play with strings and other printing commands.

Don't forget! The AMOS Competition closes soon. You could win a copy of AMOS Professional by sending in your latest creation. Programs will be judged on style, usefulness, originality and ease of operation. Your entry must be classified as Public Domain.

Send any questions to:

AMOS
PO Box 567
Mirrabeeka
WA 6061

Learn

with

Mark Little

Part 2

Welcome to the second article on the 'C' language. How did you go last time finding out what "\f", "\r" and "\t" mean? Let's check.

\f Form Feed (Clears the Screen)
\r Return (Go to the start of the line)
\t TAB - Move right to the next TAB stop

Well, that's enough home work - let's get onto something new. Most computers can access data of more than one size. This can range from a single bit up to 8 bytes or more. In the previous sample programs, there were references to "int" and "char" variables. The 'C' language generally allows operations on following data sizes:

char 1 byte (usually used for characters)
 short 2 bytes
 long 4 bytes

At least the size of "short", but may be the same as "long" - it depends on the compiler.

By using signed or unsigned numbers the following ranges are available:

Data Size	Signed	Unsigned
char	-128 to 127	0 to 255
short	-32768 to 32767	0 to 65535
long	-2147483648 to 2147483647	0 to 4294967295

For scientific applications, there are several floating point schemes. Most, except for double floating point (IEEE), use four bytes per number. Here are the more common ones and the range of numbers they can handle:

.float (FFP) 5.421E-20 to 9.233E+18
 .double (FFP) 5.421E-20 to 9.233E+18
 .float (IEEE) 3.402E-37 to 3.402E+48
 .double (IEEE) 2.222E-308 to 1.797E+308

On standard Amigas such as the 500/1000/2000, the fastest calculations use the Fast Floating Point (FFP) format software, but models such as the Amiga 3000/4000 that have a maths

co-processor (which uses the IEEE format) has super fast calculation. Some examples of number types are in the program "types.c" on the disk. I haven't included any floating point numbers, because not all compilers handle them directly, but they work in the same manner.

'C' treats a text string as an array (or sequence) of characters (type char). To use a string which will be no larger than 79 characters, I would declare the following array

```
char MyString[80];
```

The reason the array is one character larger than the expected maximum size of the text, is that 'C' adds a zero byte to the end of a string. This means any routine using the string (such as "printf"), can keep getting characters from the array until it finds a zero byte (or NULL as it is sometimes called).

Not all arrays are character arrays, it is also possible to have arrays of any data type such as floating point numbers or address pointers. The program "array" on the disk shows how to use an array. In later articles, operations on arrays using loops and other controls will make array operations much more useful.

Arrays are great when we want to group together items with the same data type. They aren't much good however, if we want to group together related information which has different data types. For example, this is useful in a payroll program. The information needed includes the employee's name (type char), employee number (type int), pay rate (type float) and so on. What could be done is to have an array for each data type.

Passing each array to the procedure would make it quite complex. This is where structures come in. A structure allows you to group different types of data into a single unit.

To make the compiler understand a structure, you need to define its contents. Below is a structure definition for the payroll data.

```
struct Employee
{
    char Name[80];
    int EmployeeNo;
    float PayRate;
};
```

To use the Employee structure, use definitions like these:

```
struct Employee Fred;
struct Employee *Pointer;
```

If you measure the size of "Fred", it would be about 88 bytes (depending on your number sizes). The size of "Pointer" on the other hand would be 4 bytes. This is because "Fred" allocates space for a copy of the structure, while "Pointer" can only hold the address of a structure.

Like any other data item, it is possible to have an array of structures. This allows the payroll program to pass a single pointer to all the employee's information. Remember from last week, a pointer must be passed to a routine if you want it to change the data.

The method of accessing an element in a structure depends on whether using a pointer or structure address. To read the value of the PayRate, you can access the structure directly by "Fred.PayRate", while you would use "Pointer->PayRate" if you were using a pointer. This is easily remembered because the arrow "points" to the structure element. An example of using structures is in "structs.c" on the coverdisk.

Structures are very flexible because you can define a structure within a structure, or you can include a pointer to the same type of structure. The structure below shows what I mean.

```
struct MyStruct
{
    struct Employee AnEmployee;
    struct MyStruct *NextEmployee;
    struct MyStruct *PreviousEmployee;
};
```


You can't, however, define a structure of the same type as you are defining in that structure. Sound complicated? That's why the compiler can't do it either. Here is an example.

```
struct YourStruct
{
    struct YourStruct *NextOne; /* A
    Pointer - Ok */
    struct YourStruct AStruct; /* Oops - No
    Good */
};
```

Let's try to calculate the size of the structure to see why this will fail. The size of YourStruct is 4 bytes (for the pointer) plus the size of YourStruct which is 4 bytes plus the size of YourStruct which is ... As you can see we never reach a result.

Each component of a complex structure can be accessed by using the ">" and "." operators together. For example, I have a pointer to a structure of the type "MyStruct". How do I find out the PayRate?

```
Pointer->AnEmployee.PayRate
```

This works because "Pointer->AnEmployee" gives the address of Employee structure. If we have the address of a structure, then we use the "." operator to look inside that structure.

Try a more complex example. How would I access the pay rate of the next employee after the next employee pointed to by my pointer? Let's solve this step by step. How do I get to the next employee?

```
PointerA = Pointer->NextEmployee;
```

and again to get to the next one:

```
PointerB = PointerA->NextEmployee;
```

Now we access the PayRate like our previous example:

```
Rate = PointerB-
>AnEmployee.PayRate;
```

We can combine all this into one line of code

```
((Pointer->NextEmployee) -
>NextEmployee) -
```

```
>AnEmployee.PayRate
<----- Pointer A ---->
<----- Pointer B ----->
<----- Rate ----->
----->
```

You don't need the brackets, but I included them to make it a bit more obvious. Depending on the structure definitions, any combination of ">" and "." operators are possible.

In the example above, the NextEmployee field was set to point to the next entry. What statement would you use to do this? There are a few ways to do this depending on whether you use a pointer. If you have problems with it, I'll give examples in the next article.

The next article also deals with control statements such as "if" and "for". Once you know how to control the flow of a program, you can start to write real programs. Have fun and 'C' you again next time.

Mark Little

Confessions

of a Modem Addict

After I had been computing for a while, one of my "friends" told me the best way to do work at home was to buy two modems. One would go in the office and the other at home. I would be able to just phone the machine at the office and get whatever I needed to do my work.

This sounded like a great idea, so I immediately went out and bought two Maestro 2400 modems from my local store. I set them up at the two locations and begun to work effectively from home "Hooray".

"THEN" after I had been happily using my modem to ring work for about three months, my "friend" told me about BBSs. Being a BBS pusher, he gave me my first number and we dialled it while he was still with me (just to make sure I did it).

We dialled the number, the phone started screaming and then the computer started asking questions. I gave all the correct answers and was told that I could call back in 24 hours and do whatever I want! Well, I rang back as early as possible the next day to be told I had "Visitors Access", this somehow translated into 30 mins per day. To my wonderous surprize I could "Download" files. Anything from odd little utilities, to pictures and actual programs. Well, I was hooked!

Every day I would ring the BBS and download as much as I could, until, after about a week, there was a message to me from the almighty SYSOP!...."I will be forced to cut your access unless you upload some files". What could I do, I didn't have any files to upload?"

Then it hit me! One of the files I had sponged was a list of other BBSs. I could just call up another one, download files

from there to give to the first one, simple! As you can imagine this worked for a while, but then my life became really hectic. Downloading and uploading became an obsession with me. I had made lists to keep track of which BBSs I was ringing and my upload/download ratio on each.

After a while I was even ringing interstate numbers in hopes of finding a new and unique file. The "File already exists" message became my nemesis, it had me tearing my hair out every time it appeared. I started to rename files just to be unique. I had become the dreaded modem junkie. It didn't bother me one little bit, I even uploaded my startup-sequence after I had renamed it. Anything to get me more time. I even went so far as to start on the international numbers (BAD mistake).

It happened, the god who rules all our worlds finally said "STOP", I got the phone bill!! I realised, with a bit of help from my bank account, that this had to stop. I tried to cut down but to no avail. I was still on the phone all the time.

Just when I had decided to try going cold turkey, the almighty telecom cut the phone off. Well there you go, now I didn't have any choice. For a while I used to use the public phone to dial BBS numbers just to hear the squeal, but then after a while I finally kicked the habit.

I have since paid the whopping great phone bill and have had my phone reconnected. I continue to use a small number of BBSs, uploading and downloading in small quantities. Even sending my own programs up to the BBS. I have become a social modem user but the spectre of the modem habit will always be with me. I hope this little story will stop someone else from falling into the same trap.

Many people who own a computer will toy at some stage with the idea of programming for enjoyment, and for the possibility of making a few dollars. It

had from magazine coverdisks (such as OZAmiga) and public domain and shareware collections (such as Fred Fish).

royalty on your work, you receive the full amount of the registration fee.

Control: You decide when and where to

So, you want to

By Perry Rosenboom

doesn't take long, however, to realise that making money out of your programming is not as easy as it sounds. The range and quality of commercially available software is stunning, and it can be a daunting task to try and compete against the giants of the software business.

Often these days the support you will receive from a so-called 'amateur' who writes shareware is better than the support you can receive from professional software houses. This could be because the person who writes the software and goes to the trouble of releasing it as shareware will usually take some pride in the work that has been done.

release your software. You have control over the cost, content and distribution.

Rewards: Much of the reward is provided by fellow Amiga users who take the time to register. Depending on your distribution efforts, these letters can (and have) come from all parts of the country and the world.



The purpose of this article is to provide a few tips to potential shareware authors, so that some of the mistakes that I made can be avoided. I'll also provide a few tips for those people who are considering sending off their registration, but are not sure about sending their money to someone unknown. This article is based on the experience of myself and a few friends, so if you have other ideas or suggestions, send a letter to the editor - we would love to hear from you!

There are some advantages and disadvantages which you will need to be aware of if you are contemplating the



Disadvantages

As the saying goes, "There's no such thing as a free lunch". If there were no disadvantages to releasing the software as shareware, everything would be released as shareware. Major considerations include:

Registration: You can't force the people who use your software to register.

Fees: If your software is released as shareware, you can't expect to ask, and get, the same amount of money that the commercially available equivalent would cost unless you have a unique product.

Support: The work is not complete when the software is released. You have a

The answer is usually found in releasing your software as Shareware. Shareware differs from 'Public Domain' in that a shareware author provides the software on a trial basis, and asks that users register and send a small donation to the author if the software proves to be useful. In exchange, you may be provided with the latest version, complete documentation, or future updates. To continue to use shareware software without registration is illegal.

There was a general view not too many years ago where shareware and public domain software was perceived as 'cheap and nasty'. Thankfully, the

release of your software as shareware. These pros and cons are not huge barriers to your impending fame and fortune, it's just that you may get there a little quicker if you are aware of them. I'll kick off with the positive side.

Advantages

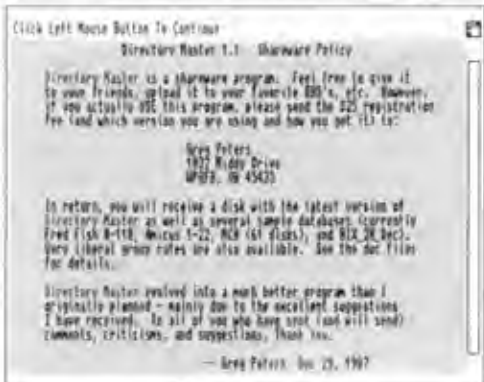
The list below is probably not exhaustive, but I think I've hit the main points:

Cost: The cost of releasing your software as shareware is relatively small when compared to the cost of releasing the same software commercially.

Copyright: You (the author) can retain the copyright to the software, instead of releasing it to the distributors.

Margin: Instead of receiving a small

moral obligation to support those people who take the time and trouble to register. If you are not prepared to support your software (or deliver what you promise) then don't release it in the first place - it



perception has changed and every Amiga user knows that there are some very good productivity, utility and entertainment software to be

gives all shareware authors a bad name. **Marketing:** As you are the only person with a vested interest in having your software released, you will need to do all

*Give people an incentive to buy. One of the mistakes I made when releasing a game called SubAttack was that there

person in the US sent me a personal cheque in \$US. I also received an international money order (again in \$US). The banks here charged me \$5 to

write Shareware ?

the associated marketing yourself. By this, I mean that it's up to you to send your contributions to PD houses, Fred Fish, or magazines. As with anything you may try to have published, don't expect everything you send in to be published by everyone to whom you send it.

Competition: Like it or not, you are competing for 'business' with commercial products. In order to be successful, the quality of your software needs to be very good.

Assuming that you haven't been turned off after reading through some of the pros and cons, I've listed a few tips that I've learned the hard way over the last couple of years. These are not the secret to instant success (I'm not giving up my

was no real reason for some people to register. The game is fully functional, and if users don't want the next version,



or the source code, they will just use the shareware version without registration. This is illegal, but obviously it's difficult to enforce. At the time of writing this article, SubAttack is sitting at position 75 in the top 100 PD releases according to a particular UK based magazine. I've not had a single registration from the UK!

With the shareware version of Solitaire which was released in the December issue of OZAmiga, you will have noticed that some of the features are disabled. Also, there is an annoying message that flashes up at random. This means that people who register will receive something immediately that has more functionality than the shareware version. Many demonstration releases of commercial software use the same principle. For a database, you may only

change them, and I was left with \$5. You may need to remember this and set a scale of registration fees for different countries to take into consideration the fees incurred.

*Personalise your registered copies. This takes a little effort, but it's worth it. If you registered for December issue of Solitaire, then you would have received a copy of the game with your own name included in the title screen and in the "About" screen. People are less inclined to give away your software if their own name appears in it! Large software companies (including MicroSoft) use this method to try and protect their software. Just because you are small time doesn't mean you can't use some of the principles adopted by the commercial software manufacturers.



day job just yet!), but they may help:

*Put a reasonable price on your work. You have a better chance of a sale if people think that it's good value. That doesn't mean that you need to give your work away. Take a critical look at your own work and ask yourself how much you would be prepared to pay for it. Remember that you are in competition with commercial developers, other shareware authors and (unfortunately) pirates.

*Do your best to produce good quality, bug free, software. Your best chance of exposure and registrations is for software which looks good and works the way it should. We all know that no software is totally bug free, however putting the extra bit of effort in will produce rewards.

*Provide some decent documentation. There's nothing worse than getting a great looking piece of software for a good price, and not knowing how to use it properly due to poor (or no) documentation.

be able to save 20 records. For a word processor, you can only save a short document, and so on.

*Make it easy for people to buy. With my early releases of shareware, I requested Aussie dollars because our banks and governments can act like robbers. This may have put some people off registration - I'll never know! One



*Make it easy on yourself. What I did in the source code for Solitaire was include a variable which is tested when the program is running. If the variable is set to a certain value, the game is registered and all features are available. Otherwise it plays as shareware. Sure, there is some extra code sitting there, but that extra code makes it very easy to produce a shareware or registered version (all I need to do is compile it). It also means I'm only supporting one version of the code!

*Write the code for NTSC. The US market is a big one if you can crack it. My game of SubAttack was written for PAL (which is the standard used here and in Europe), and I've had letters from the US requesting an NTSC version (which I have now completed). This explains why Solitaire was written for NTSC. On PAL systems, you'll see a black space at the bottom of the screen. Some developers put nasty comments in there about NTSC systems...



*Include appropriate disclaimers, copyright and shareware messages. It's important that you and the users of your software realise that shareware is NOT public domain. You still own the rights to shareware code, and people have an obligation to send you some cash if they plan to use the program. If you don't tell them, then they won't do it.

If you don't copyright it (just stating the copyright in the documentation and software is usually enough) then your hard work could appear anywhere, or on sale somewhere, and you can't do anything about it. Remember - you did all the hard work! If you are not sure what to include, then take a look at some other shareware.

*Be prompt. If you get a letter or registration, answer it as soon as you can. Customers are hard to come by - you can't afford to annoy them! If people take the time to make suggestions, then acknowledge them. If the suggestions cannot be included in the next version of your software, then explain why not.

*Don't expect to get rich. If you are just doing it for money, then you'll be disappointed.

*Distribute the software as far and wide as possible. It's fine to give a copy to your local PD house or computer shop, but only the people in your area will buy from them. If you want the best chance of getting a return for your work, you should consider going International. Most software houses get their public domain and shareware software from the

Fred Fish collection. Any of the latest Fish disks will have details on how to send stuff to him. Just through Fish disks, I've had letters from 6 different countries!

*Consider a Post Office box. I use one which I share with a co-author, Andrew Kreibich. The cost is currently around \$30 per year, and it's well worth it. I've moved house once since releasing my first two games, and I'll be moving again within 12 months of writing this article. If I was using my home address, there



would be games around all over the world with old addresses!

*Be patient - it takes time both for the letters to start coming in, and it takes time to send registered copies (and letters) back out. It's something to bear in mind.

For those of you who are considering registering for shareware, there are also a few things to think about:

*Shareware authors do it mainly for fun, and are part-timers. If this means a couple of weeks delay in getting a reply,

please be patient. Shareware authors are Amiga users just like you!

*You have to trust the Shareware authors to deliver what they promise. I've endeavoured to be honest and upfront with my 'customers' at all times. In one case even sent back the registration fee because I knew that I couldn't deliver what I was being asked.

*Ask and ye shall receive (well, maybe). If you are interested in programming and would like the source code, ask the author and state your intention. The author may tell you to get stuffed (which is unlikely) or you may in fact receive what you asked for. I received a letter and registration from someone who requested my deck of cards from Solitaire for inclusion in another AMOS game (Hi Phil!). I was more than happy to send them over, especially as he had included a disk. I'm trusting that person to acknowledge my contribution if he releases his software - I haven't been disappointed yet.

Well - that's it. I hope I've been of help, and that I haven't put anyone off. There's a lot of satisfaction in seeing your work on an OZAmiga CoverDisk or in an ad for a PD house. It also helps if you have an understanding family, as I do. If you would like to contact me for any reason, then please write to:

**Perry Rosenboom
POBox 333
Wantirna South
Victoria 3152.**

PRESS

Rockford, Illinois, March 1993.

ICD Incorporated, a leading designer and manufacturer of Amiga hardware enhancements, today announced a new standard for the Amiga 1200 computer.

ICD's new Viper 1230 was designed to give the popular Amiga 1200 the performance of a workstation. Viper 1230 offers the A1200 owner a 68030 accelerator supporting high speed memory expansion, an FPU coprocessor socket, a battery backed-up real-time clock and a unique 16-bit Direct Memory Access (DMA) port for further expansion capabilities.

Viper 1230 uses the power of the Motorola 68030 and supports both EC and MMU versions from 40 to 50 Mhz. When compared to the A1200's stock 68EC020 running at 14 Mhz, the larger cache and higher speed of the Viper 1230 processor will really make applications fly.

Up to 64 MB of fast RAM can easily be added to Viper 1230 using industry standard 32-bit wide 72-pin SIMM modules. Two high quality SIMM sock-

ets are on board for memory expansion. Burst mode, for top speed, is fully supported using low cost, page mode DRAM.

With the addition of a high speed Motorola 68882 math coprocessor (FPU), all floating point math routines will run at near warp speed. Applications which rely heavily on floating point routines such as animation, ray tracing, image processing, DTP and CAD will show an amazing improvement.

Viper's DMA port (VDP) allows many opportunities for high speed add-ons of the future. Products like an SCSI-2 controller, DSP board, modem or networking card could be developed to accommodate this port. VDP specifications are published in the Viper 1230 hardware installation manual.

Viper 1230 was clearly designed with the customer in mind. The RAM, CPU, FPU and the clock battery are all socketed and changable using industry standard parts. The low-cost 40 Mhz 68EC030 is supported as well as the 50 Mhz 68030 for those who want the ultimate in performance with an MMU. Both 40 and 50 Mhz FPUs are supported. Low-cost industry standard SIMMs are used for memory expansion. Viper 1230 and VDP boards are easily installed without removing the top of the computer. This ensures that Commodore's warranty remains intact.

I will publish the Australian distributor information as soon as it becomes available.

**THE BIGGEST PD
Collection in Oz!!**

**Public Domain
from only 50c**

AMILLIGHT 100% Amiga



***Purchase any \$25 Pack
and get 5 minutes in***

VIRTUAL REALITY FREE!!

THE BIGGEST
Over 60 different Libraries,
Thousands of new files
imported and downloaded
from all over the planet
(including 17Bit)

Gigabytes of new AGA Graphics!

(09) 367 4482

4 Disk Catalogue \$4
\$1 Post & Packaging
Mention this add when you order
to be in the draw for 1 full hour
for two people to battle it out in
CYBERSPACE

Corner Angelo & Tate St.
South Perth
West Australia

PO Box 233
Como 6152
West Australia

PHONE ORDERS WELCOME



THE BEST

Only High Quality

Nashua disks used

Free Help & Advice

Same day air dispatch

Public Domain from 50c

CYBERZONE ACCESS FREE!!

(09) 367 4482

I have had many requests for a section that deals with the basics, how to do those things that the "Average" computer user takes for granted. So I found two people that knew little more than how to turn it on and play a game and I taught them to use their computers much more effectively. Out of this comes the following series of tutorials. For the tutorials to be effective you will need to have a play-around with each command as we go along.

My first problem was deciding where to begin. I must assume that everyone knows how to use their mouse to select icons with the left button and menu items with the right. These are some of the first topics covered in the manual so if you're not sure, have a quick read.

The first area I will look at is Workbench. There are a couple of different versions, each with differing abilities. I will use V1.3 and V2.04 as these give the best overall view (plus they are the two versions I have!). In future articles I will cover preferences, the startup-sequence, the workbench programs, setting up a hard disk, using PD utilities and much more so I hope you can join me.

Well, let's get started! I will go through the version 1.3 commands first and then go on to the version 2.04 commands. So if you are a version 2 user you may wish to jump forward to that section.

When you plug in your Workbench disk, the program loads and you are soon confronted by the Workbench Screen. From here you can open windows, run programs, copy disks and more, but then that is what I'm here to show you isn't it. Your V1.3 workbench has three pulldown menus called Workbench, Disk and Special. In these menus are a number of useful commands, below is a description of each. Make a back up (copy) and experiment. We make a back up because it is not good to delete files from your original disks.

WORKBENCH

- **Open** If you click on an icon, then select Open, it does the same thing as double clicking on that icon. eg. it will open the drawer or execute the file associated with that icon.

- **Close** This one is pretty self explanatory, it does the opposite of open.

- **Duplicate** If a disk icon is highlighted when you select this menu item, you will perform a disk copy. Where as, if a file or program icon is selected, that file or program will be duplicated. The same result can be achieved by dragging the relevant icon onto the destination icon.

For the New User.

- **Rename** After we have duplicated a disk or file it is called "copy of XXXXX.xxx". So what we do is rename it to something more suitable to our purposes.

- **Info** This item will open a window to display various information about the selected icon. The information differs a bit depending on the tool types.

- **Discard** Does exactly what you might think. It will not allow you to discard a disk but all files are fair game.

- **Empty Trash** On the Workbench disk and many others you will find an icon called "Trashcan", use it to put unwanted files in. When you have finished but before you pack up, you should empty the trash, this effectively deletes all files in the trashcan.

DISK

- **Initialise** Another term for "Format a disk", it sets up a disk so that the Amiga can use it. A blank formatted disk will hold approximately 880k of files.

- **Clean Up** This will rearrange all of the icons in a window so that all are displayed in an orderly fashion.

- **Last Error** As you may have guessed, this will display the last error message.

- **Redraw** If you have made any changes to your workbench from, say the CLI, they are not immediately visible on screen. That is why we use Redraw.

- **Snapshot** Use this command to set your windows up so that they come up the way you want them to. If you open a window and move all of the icons to your preferred position, then select all of them (click on each whilst holding down the "Shift" key) and Snapshot, they will appear in that position every time you open that window. If you Snapshot a highlighted window, it will also open in the same position each time.

SPECIAL

- **Version** Will display information about your Workbench version and date of release.

Before I go on to the additional commands in V2.04, I would like to

recommend to all V1.3 users to upgrade to V2.04 or better as soon as possible as it adds a whole new dimension to what you do. When I was still using V1.3 I found I had a lot of extra utilities to do the things that V2. makes easy.

The first difference we find with WB2 is that it has four pull down menus instead of two. The menus are called Workbench, Window, Icons and Tools. There are quite a few additional commands, so I will get right down to it.

- **Backdrop** This causes your workbench window to disappear and your disk icons to be displayed on the basic Workbench screen. Doing this can be very helpful when you have multiple windows open it saves having to keep moving the workbench window out of the way when flicking through windows.

- **Execute Command** This opens a requester which lets you execute an AmigaDOS command without having to open a Shell.

- **Redraw and Update All** Both commands do much the same as Redraw on V1.3.

- **About** Again, this will display the version and release date information.

- **Last Message** Sometimes you will see a message flash across the top of the screen but not be quick enough to read it, so you would look at Last Message.

- **New Drawer** As it's name suggests, this command will create a new drawer and ask you to rename it.

- **Open Parent** Selecting this option will open the parent of and highlighted window and bring it to the front.

- **Close** This will remove the selected window from the screen. It is the same as clicking on the close window gadget in the top left corner of the window.





Select Contents
Will highlight all icons in a selected window, just the same as holding the Shift key and clicking on each one.



Clean Up
Operates exactly the same as under V1.3.



Snapshot When you highlight this menu item a SUB-MENU appears with two items displayed.



Window will save the position and size of a selected window.

All will also save the position of all the icons within that window.



Show Not every file on a disk has a corresponding icon, so the show command allows you to select either of it's sub-menu items.

Only Icons will obviously display only those files with an icon attached.

All Files on the other hand will display every single file on the disk.



View By This allows, via a four part sub-menu, files to be shown by either Icon, Name, Date or Size. When used in conjunction with the show commands you can set your windows up and use them similar to a file utility.



Open Performs the same function ascribed to it in earlier versions.



Copy This command will let you copy files as well as disks.



Rename Another command that hasn't changed much.



Information The icon information supplied by this command includes Name, Image or icon picture, Size, Stack, Status and Last changed date. If the icon represents a drawer, project or tool, there is an additional set of six selectable attributes.

Script when selected, tells the computer to execute a script or batch file (a text file of AmigaDOS commands).

Archived is set by some of the back up utilities to let you know that the file has been stored or archived.

Readable makes the file available to read or access the information in the file.

Writeable means simply that, you can add information to the file. If Readable is selected and Writeable isn't, you will be able to look at the file but you won't be able to change it.

Executable means that you can execute or run the project or tool associated with that icon.

Deletable is fairly obvious to most, if it is unselected it acts as a minory form of protection for the file.



Snapshot Selected from this menu, it will save the position of the icons instead of the window.



UnSnapshot Sticks out like a sore thumb eh! I find it easier to just re-snapshot over the top of the old configuration.



Leave Out If you have a particular file that you use every day but you have to open a disk icon and a couple of drawers to get it, you can Leave it Out. This will move the icon from it's original window onto the workbench window. It will not move the associated file, just the icon, but it does

make it possible to use the file straight from workbench. The icon will stay there even if you reboot.



Put Away When you have used your left out file as much as you needed to, you should put it away to keep unused icons off of your workbench.



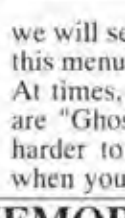
Delete This one is easy, it will delete any selected icons.



Format Disk This will open a format requester that asks for a new volume name, whether you want a trashcan or not and other disk type information. It is quite quick and you can use more than one at a time.



Empty Trash Same as in V1.3.



Reset WB Does exactly that.

Later on when we look at using PD and other utilities we will see many other commands under this menu. At times, many of the menu commands are "Ghosted", this means that they are harder to read and are not highlighted when you point at them with the mouse.

MEMORY EXPANSION PRICES at June 1st 1993

MEMORY		DRIVES	
1MBx1 - 80ns	6.25	SEAGATE 2.5" I.D.E	
4x256 - 80ns	6.25	64 MB 16ms 32K	240.00
1MBx4 ZIP - 70	26.00	85 MB 16ms 64K	380.00
1MBx4 ST COL - 70	26.00	128 MB 16ms 64K	525.00
1MBx4 PAGE - 70	34.00	209 MB 16ms 64K	865.00
1MBx8 - 80 Si	49.00	SEAGATE 3.5" SCSI Int.	
4MBx8 - 80 Si	205.00	248 MB 72ms 128K	563.00
4MBx - 60 - 64	268.00	452 MB 12ms 128K	1480.00
4MBx32 - 70 - 72	225.00	ROCTER FLOPPY DRIVES	
PCMCIA V2 - 1M	260.00	FD386 3.5" Ext 880K	116.00
PCMCIA V2 - 2M	450.00		

Please phone for the latest prices. Sales tax 20%.
Overnight delivery, credit cards welcome.

Pelham PTY LTD

Tel: (02) 980 6988 Fax: (02) 980 6991
1st floor, 100 Yarrara Rd, Pennant Hills 2120.
PO Box 382, Pennant Hills 2120

The reason they are ghosted is, in most cases, that something on the workbench screen must be selected in order for that operation to be performed. In my next installment we will look at "Preferences" and how to set up your machine to suit you.



VLAB

Real Time Video Digitizer

Reviewed by Mark Scholmann

The Amiga Computer, from the first A1000, has always been a good "video compatible" computer. The Amiga's screen display is made for PAL video output. We can start with a \$60 encoder, up to a \$4000 broadcast quality genlock. The newer models, the Amiga 600 & 1200's, have composite video output as a standard feature, making output even easier. When it comes to getting video images into the Amiga, we have had to work hard with colour wheels, RGB splitters, or put up with low quality images. That is until now! Ok, before you all say, "What about Digiview?", well I'm not knocking it, but you still have to work hard for a colour image (even if they are high quality!). Other "frame-grabber" devices have been around, but nothing that I have seen has the quality and price value of VLab.

What is VLab?

VLab is a 24 Bit full-frame video digitizer. What this means broken down is:

- 1) The digitizing process is in 16.7 million colours on the VLab board.
- 2) Full-Frame means that VLab can digitize a full video frame in 1/25th of a second.
- 3) The digitizer part of the VLab converts the analogue video image into digital form for the computer to process.

When you open up the VLab box (with its very glossy cover), you will find a "card-slot" plug in board for the 2000/3000/4000 series, a 150 page manual, and an install disk. To keep all Amiga users happy, there is a version with a parallel cable connection for the 500/600 & 1200 Amigas. Looking at the board, you will notice two composite (VHS) connections at the end of the board.

These are for two video inputs, which are selectable through the control software. For high-end users, there is also a Y/C (S-VHS or Hi-8) version of VLab as well. So, with the computer disconnected, I inserted the board into one of the expansion slots, and connected my composite lead from a VCR to one of the input connectors (they are RCA type connectors). Now for the manual. Flicking through the manual reveals a very detailed reference to the control software. Every menu option is explained in an organised fashion, with an easy to find contents section. So it is quite easy to sort out any problems, if they arise. There is no tutorial section, just detailed explanation of the technical aspects of video digitizing.

Installing VLab's hardware and software only takes a few minutes, so you can start digitizing straight away.

The first time I used the VLab control software, I found a blank screen, with a title bar containing information, and a small window with several selectable options. One "button" was called START, so I pressed it and a requester appeared, stating "NO VIDEO SOURCE DETECTED". OK, so we're not going to be starting straight away. Then I remembered that there are two inputs, so I looked in the menus and found the Define Source option. A window appeared showing many controls over the video input signal. You can name each piece of video gear you expect to be digitizing from and set specific parameters for each one. Then when you switch between devices or tapes, you can easily reset all the digitizing parameters by choosing the pre-defined name from a simple list. You can set factors such as: the size of the image, the X & Y offset from the top left of the video frame (portions of the

screen that are normally hidden from view on a TV), the RCA connector position on the board, then there are advanced settings - signal filters, PAL/NTSC, VCR or camera and noise filtering. Setting the VCR button on, activates the VLab's built-in time base corrector to stabilize any incoming images. This is handy when the quality of the recording on tape or the player is a bit suspect! I set up my requirements and called it StandardVHS as a reference. I then went to the Colour Correction menu option, which opens a window with slider bars, controlling the RGB colour values, contrast, luminance and gamma corrections. I increased the gamma control several points, so that my slightly dark video, would show up a bit brighter, yet hold its relative contrast levels. Now I was ready to try again, so I selected the Start button



again, and this time with a short delay, I watched the icon flip through several icon shapes, and finally it stopped and nothing appeared on the screen. Oh well, it grabbed an image at least, now to find out how to see it!! Well, another look through the menus reveals the Preview menu options. First I activated the Auto Preview mode, so after an image is digitized it will show on the screen. (Otherwise I could simply press the PREVIEW button next to the Start button). I also set the preview screen to the current background, as a 16 grey scale image. OK, let's try again. I pressed the Start button, and after some waiting a grey scale image of my video appeared on the screen. Yeah!

So, you're probably wondering what all this waiting is for, if it is supposed to be a real-time 1/50 sec frame grabber. Well, it is, that's how the fast the board can "grab" an image... getting it into the Amiga is a different



(and longer) process. One frame of video in 24 Bit quality as a 640x512 pixel image can take about 1 megabyte of storage space, so the computer needs a bit of time to read this off the board. The benefit is the image can be a fast moving racing car, people running past, to a stationary flower, it will "freeze" anything instantly.

Another option in the menus, is the Monitor function: this allows a small window on the screen to show a slow frame rate image of the incoming video. In the large window mode, I reached 6 frames per second which is fine, if you are working out the framing of an object through a camera, or just need to see where you are on a video. I do recommend a TV to watch your video or camera through, as you will see the true colours and exact point on the tape, especially if you are doing continuous "freeze-frame" digitizing. I realized that a lot of people may only use their images in lo-res (320x256), so I set up my "scan" size, and started my first proper scan. Which really is only seconds away! I then used the Convert menu option, to convert the 24 Bit "grab" into a Ham8 (262,000 colour mode on the new AGA machines), which took awhile. This was slower than other image processing packages I have used, but I found out later, Vlab was doing several special filtering techniques on the images, which were set as defaults in the program. Once I saw that the image looked great in Ham8, I continued on, and grabbed the rest of the images you can see below. I needed them in 24Bit so I didn't go through and convert them all, but in normal use of Vlab I would.

To cover some other important features, I'll briefly list down some ideas and ways of using VLab.

Vlab Possibilities

For people with the budget and needs, you could get the S-VHS Vlab for clearer images, with simple Arexx scripting you could drive the VLab through Art Department Professional, to do all sorts of automatic image processing. (The Art Dept. loaders and savers are provided standard with Vlab). Another use could be for time lapse digitizing. Using VLab's built in Sequence "grab" option, you can set up VLab to "grab" an image every 10 seconds, for 100 images, and have them save to disk as a special compressed format. Again, more control could be gained from your

own Arexx scripting, for time controlled or "event" controlled "grabbing". How about some image detection possibilities, by using VLab to "grab" an image, then convert to 2 colours, and using custom software to determine where an object is!

Or for textures and object wraps in 3D rendering packages, VLab can make 24Bit files (as standard), so that your renders are of the highest quality. This could be using lo-res "grabs" (as textures don't normally need to be too large), and this has the benefit of faster digitizing. How about all these great morphing packages coming out... use the VLab to capture faces and objects in 24Bit quality.

So to sum up my time with VLab... It is a very high quality digitizing product. I have managed to digitize

some sequenced segments from videos (frame-by-frame), and have had good comments about the results. I use Art Dept. a lot, so I now just use the VLab loader from there, as I can then manipulate the images with greater control, for my work requirements. The VLab supplied software handles more detailed features over the digitizing process than Art Dept., and I would definitely recommend it's use for those wanting to push this board to its limits.

You can contact me at work for more information!

Mark Scholmann

HEADLAM 09 388 3666
COMPUTERS

VIDEO FRAMES... (The following is meant as a general guideline only!)

When watching television, you are looking at 25 images (frames) per second. One frame is made up of two separately recorded parts (called interlaced fields), one field contains the horizontal lines 1,3,5,7,9,11... down the screen. The other field contains every other line, 2,4,6,8,10... When these two fields are interlaced, they form one frame. To "draw" these fields onto the screen, the TV "draws" one field, then returns to the top, and "draws" the other field into position. Doing this really really fast, we see 25 frames (50 interlaced fields) per second.

Your eyes need a finite time before they register "light" images, so when the electron beam in a TV "draws" the image onto the phosphor screen, you see a complete image, and not the single point of light, that it is! The image "drawn" in phosphor on the TV starts fading away, while new fields are continually "re-drawn". With our eyes, we see colours fading and reappearing which in certain combinations create a "flickering" effect. Computers are very exact, and often (compared to real life objects), have very sharp edges. There are ways to avoid the interlace "flicker"... but that's another story!



A Full-Frame "grab" of a kick boxers' fast moving leg, reveals how the picture was recorded over two **Fields**, as separate images.



The great Australian rip off.

NOTE: The views and comments expressed in this article in no way represent the views of OZAmiga and/or the owners of OZAmiga. Views and opinions are not designed to be libelous or defamatory and are written in the interest of public information.

**By:
Bill
Holder**

Ever wondered why those prices in overseas computer magazines seem to be a hell of a lot cheaper than your local computer store is offering? Wondered why you haven't seen the latest game everyone overseas is playing? Or wondered why the Golden Oldie in your latest overseas magazine is one you never saw on the shelf at your local computer store?

I'll tell you why in two simple words - Software Suppliers.

And why is it their fault, Closed Market.

"The price of software is too high!"

What this means is that a few people decide what you will and won't get to choose from at your local store, regardless of whether it is available overseas or not. If they don't want to import it you won't get to see it. How do they get away with this? A mixture of stand over tactics and legal jargon.

push off. I spoke to retailers and heard some very disturbing stories about the antics of suppliers. I cannot mention the retailers by name due to fear of repercussions.

One of the questions I asked of suppliers was "Why the high prices?". This got me the usual response of either low Australian dollar or the classic "Piracy robs the industry of income so we have to increase the price to compensate." Oh yeah, then explain why console cartridges are so expensive. I know of people who purchase software by mail order from overseas. Although they pay retail price, shipping and insurance, have to worry about the fluctuating Aussie dollar, they are still paying a great deal less than what is offered locally.

One supplier even told me software is being re-released under a budget label sooner than it used to be. Not true. The amount of software available under budget labels is bigger than it was before but the software being released is only the stuff that didn't sell. If it was a lemon at full price it will still be a lemon at half price. I have yet to see a popular

more likely a game from his selection of new software. Now this all seems like a good idea to you and me but it is in the same league as murder to suppliers. Nothing can be proven but it is rumoured that the reason why the store in question has trouble getting new stock is the suppliers have the shop black banned. Stock does arrive but it is always late. Another West Australian soft-

A PIRATE is nothing more than a common thief!

ware retailer tells the story of selling a second hand software package and within half an hour getting a phone call from a supplier in the Eastern States threatening him about his second hand software dealing. Spies in the software industry? You bet. Ask any computer retailer how many calls they get asking for this program or that program when it is not available locally but is available overseas. It is estimated that only seventy percent of these calls are genuine customers while the rest are calls trying to catch a store importing it's own software from overseas.

As an example of how restricted our selection is I ordered some software from overseas at the beginning of the year. Great titles like Curse of Enchantia and Trolls. Sure Curse of Enchantia is out now but where is Trolls? Another example last month I was sent Sleepwalkers and Superfrog by a friend of mine in England? Ask your local retailer has he heard of them. I think not. They are old overseas and unheard of here.

I used to think it was just me but even the Federal Government has said enough is enough and has plans to break the monopoly by the end of the year, which means the local store can get software from wherever they want. The sooner the better in my opinion.

Next edition we cover piracy and the new fad - freaking.

Bill

For Fast Efficient Service

Perth Commodore & Amiga User Group

U3, 18 Crocker Drive, Malaga WA 6062

Repairs Free Estimates Large Amiga PD Lib

Send SSAE for free list of available hardware and software.

Disks at \$1.00 ea.

Auto Mouse + Joy Switch	\$49.95
A500/A2000 RomSwitch	\$29.95
Super Arcade Joystick	\$60.00

Phone: (09) 249 6559
Fax: (09) 249 5495

Somewhere out there is an employer just looking for a programmer that can get a real time rotating cube, light sourced with scrolling text on each face.

game released as a budget game in less than eighteen months.

Now for the dark side of the Suppliers. A West Australian Computer game dealer has been threatened with legal action so many times he has lost count. Even though his business is legal the Software Suppliers regularly threaten him with legal action if he doesn't stop. His big crime? Dealing in second hand software. He doesn't even offer cash for your old game but credit in his store, so you have to buy something else from the store, either another second hand game or

I contacted many of the Software Suppliers and was either told to push off or fed a load of garbage that made me want to

I was sitting at my trusty A500 the other day, dreaming of the A4000 I am going to buy when I saw this list concerning compatibility problems the A1200 and the A4000 had with existing software. This list is compiled by a person called Asha Develder - good one Asha!!!

Some have major problems, some will not work while others are fine. This list is in no way complete so if you know of any not on the list drop me a line here at OZAmiga or NETMail me and I'll keep this list as up to date as possible. I'll also try to get news of bug fixes and versions of programs that are rewritten for the A1200 and the A4000.

Programs that run with no problems

1869 (old)	Eye Of the	Police Quest II
A-Train	Beholder I&II	Ports of Call
Agony	F-15 Strike Eagle	Push Over
Air Support	Fast Break	Railroad Tycoon
Amnios	Gem'X	Red Zone
Antheads	Gem'Z	Resolution 101
Apydia	Gunship 2000	Secret Of Monkey
Aquaventura	Guy Spy	Island I & II
Archipelagos	HeatWave	Sensible Soccer
Art Department Pro 2	Hexuma (old)	Shoot 'Em Up
Arthur:	Horror Z	Construction Kit
Quest For Excalibur	Indiana Jones IV	ShufflePuck Cafe'
Atomino	Indy 500	Silent Service
Awesome	International Karate +	Sim City
Barbarian II	It came from the desert	Space Ace I & II
Bard's Tale III	Jonathan	Special Forces
Batman	Jocky Wilson Darts	Star Glider II
Battle Chess	Killing Game Show	Surgeon, The
BC Kid	Larry V	Teenage Mutant
Beast III	Leander	Ninja Turtles
Beyond The Ice Palace	Lemmings (+ DataDisk)	Test Drive II
Boxing Man	Lionheart	ThinkCross
Bundesliga Manager	Liverpool	ThunderStrike
Carrier Command	Lost Dutchman Mine	Toyota
Cash	Lost Patrol	Ultima VI
Chaos Engine, The	Magic Pockets	Uninvited
Civilization (old)	Maniac Mansion	Volfied
Classic Invaders	Midwinter II	Waxworks
Cool Croc Twins	NickFaldo's	Who Framed Roger
Cool World	Championship Golf	Rabbit
CribbageKing/Gin King	Nigel Mansell's World	Wing Commander (old)
Cytron	Ork	Wings
Dragon's Lair II	Pinball Dreams	Wonderland
Dune	Pinball Fantasies	Wrath of the Demon
Epic	Plan 9 From	
	Outer Space	

Special AGA versions (A1200, A4000)

1869
Civilization
Cool World
Hexuma
Lethal Weapon
Nigel Mansell's World..
Sleepwalker
Wing Commander
Zool
Art Department
Deluxe Paint IV - AGA
Personal Paint

Rick Dangerous
Robocop 3
Shadowlands
Shoe People
Speedball 2
Street Rods 1 & 2
Supercars
Supremacy
The Games: Summer Ed.
The Games Winter Ed.
Thundercats
Utopia
VectorBall
Venus

Games that will run with restrictions.

Restrictions can be: (C) Disable CPU-Cashes (E) Chips in ECS-Mode or just worse graphics, less/no sound.	Cool World (old) Das schwarze Auge Dyna Blasters Elite (C&E) Espana: The Games '92 Formula 1 GP History Line (E) Hook Indy III (C) Interceptor F/A-18 Jaguar XJ 220 (C) Kick Off 2 (C) Kid Gloves (C) Lethal Weapon (old)	Lord Of The Rings Lotus (C) Pacific Islands Parasol Stars Pegasus (C) Pirates! Populous II (C&E) Race Drivin' Red Baron Robosport RVF Honda (C) Sim Ant Sim Earth Steigenberger Hotel.. Terminator 2
688 Attack Sub Addams Family American Football Battle Command BSS Jane Seymour (C) Campaign Carl Lewis Challenge CarrierCommand (C&E)		

Thunderhawk (C)
Turrican
TV Sports Football (runs fast)
Vroom (C&E)
Wizkid
Zool (old) (C)

Games that wont run.

Alien Breed
Amberstar
Another World
Armour Geddon
Beast II
Bitmap Compilation I
Black Cauldron
Blood Money
Cadaver
Carthage
Castles
Chip's Challenge
Chronoquest II
Colorado
Corporation
Dr. Doom's Revenge
Dragons Of Flame
Dream Zone
Elvira II
Eskimo Games
Fighter Bomber
Fighter Duel Pro
Final Assault
Fire and Ice
Grand Monster Slam
Grand Prix Circuit
Harlequin
Heimdall
Hudson Hawk
Impossamole
Into The Eagle's Nest
James Bond
Logical
Loom
Lotus Esprit Turbo
Lotus 2 & 3
Overlord
Pac Mania
Populous
Powermonger
R-Type II

Cover Disk # 7

NCOMM V2.0



This issue we have included a comms program to compliment Raff Lerro's communication and technical column.

NComm V2.0

is a shareware comms program that I have found very reliable with all Amigas. The NComm interface is easy to use and configure. Raff will help you configure NComm in the next issue, but have a go yourself before then. (The phone book included has a number of good Bulletin Boards for you to call with your modem.)



VIRUS CHECKER V6.25

Again we have included an update version of Virus

Checker for you to check your disk collection for all the latest viri. We cannot be to careful about the spread of any virus!

For those that requested knowledge on how to alter your startup-sequence to make the Virus Checker, load on startup and check all incoming disks, you should add this line near the end of the file, preferably after the loadWB command.

<path to VC> Virus_Checker

LHA V1.48e



I believe LHA is the most readily used and supported archive program, so I have included this archiver

for you to extract the files that you download from your local BBS with NComm. To find out how to use LHA; please consult Raff's column in this issue.

JARGON DICTIONARY

We have decide to place the OZAmiga Jargon Dictionary on the cover disk to save room in the magazine proper. Until I run out of Jargon, this dictionary will be updated before every issue; so if you ever hear a word or phrase that you don't understand; remember the OZAmiga Jargon Dictionary.

I hope our cover disks include programs our readers wish to use. Unfortunately we have not recieved many Aussie programs from our Australian programmers. If there are any programmers out there that wish for their programs to be placed on the cover disk; please send the programs to PO BOX 188 Southport 4215 QLD. Any suggestions for the cover disk will also be welcomed.

Regards Juan



W.O.C. London

By
Sandy
Stevens

Whilst in England recently I managed to get along to the 'Amiga Format Live 93' show which was held at the Wembley Exhibition Centre, right along side Wembley Stadium. There were upwards of fifty exhibitors showing their wares to an ever increasing crowd of avid Amigans. Unfortunately I found there to be more retailers there to sell their products than there were those displaying new and exciting products.

Of the new products on display only a few really rate a mention. The most noteworthy of which is the AMI VGA adaptor and software driver. Allowing you to simply plug it into the video port, load the software and obtain full VGA display. This unit is



Tony Keefe
Show Organiser

compatible with all versions of workbench and will be utilised by many serious users.

Digital International showed off their newest release, Wordsworth V2 which is for machines with the AGA chips (A1200, A4000). They were also pushing a whole new range

of fonts and clip art to go along with it.

Electronic Arts have used the show to run interactive tutorials on the Deluxe Music Construction Set V2.0. Simple enough for a beginner and powerful enough for the professional musician, this program takes the drudgery out of music by handling things like transpositions and sheet music. Now supporting 48 staves of music, this is bound to be a welcome addition to the Australian music scene.

Of course some of these products may not be available here for some time, and some possibly not at all. So don't hassle the local retailer just yet.

It is also interesting to note two other products being given big billing at the show were Directory Opus 4, which has been out over here for a while and Miracle, the piano teaching system which was released here at the WOC show in 1991.

I'm told that the best Amiga show held in that part of the world is in Germany. This may explain the lack of new product displays put on in London.

Over the WIRE

Archiving



with
Raff Lero

Modems for The BBS User Part 4

Hi, and welcome. In this, part four of my series, I hope to explain some of the mysteries of Archival programs and to explain their uses.

What is an Archival Program?

Well, quite simply, it is a program, or group of programs which will allow us to COMPRESS and/or STORE our software or text in a single file.

The reasoning behind this as far as we the BBS user is concerned, is fairly simple. As was mentioned in one of my previous articles, compressed programs are smaller and therefore take up less disk space, and also take less time to up or download. The process of storing files, be they compressed or not, is also quite handy, as having one archive containing many files and sometimes sub-directories is very handy.

Imagine this as an example, you have just completed a ripper new program, but the program needs a certain directory structure containing certain files in certain directories in order to function correctly. How best to store this setup, so that the end user doesn't have to worry about a complex setting up procedure? Well you could give the end user all of the files, and a text file called "read me" or similar to describe how to set up all of these directories etc. But wouldn't it be easier if we had a trusty archival program to store ALL of your files, COMPLETE with their DIRECTORY STRUCTURE, in ONE archived file?

You bet it would, and as well, our programs etc could be compressed also.

There is another type of archive that is handy to use, and that is one of the WHOLE DISK archivers/compressors. With a program of this type, LHWARP and DMS are good examples, you can

actually store a COMPLETE DISK in a single file, Dos or NonDos, bootblock and all. They, like the archive programs for files, have many options for their usage and are quite versatile.

One of the first archive programs to come along was called ARC. It is in limited use these days as there are many better programs kicking about but it is still worth mentioning. Many of the newer programs were spawned from this program.

LHarc is one such program and it is considered by many to be the best of the archival programs to date. It has a good compression ratio and is quite quick, even on standard unaccelerated machines.

LHarc has numerous commands and options and these may be viewed via the CLI, if you have LHarc in your current path, by typing LHarc on the command line without any commands or switches. For those of you who can't do this, I will give a short preview of the most common commands. These commands most probably work with ARC but I cannot verify this. My version of ARC must have taken a bait as it produces a lockup on my A2000.

The command template for LHarc looks like this.

```
Lha [-<options>] <command>  
<archive[.lzh/lha/run]> [[<homedir>  
<file spec.>] [<file>] [<destdir>]
```

Where <options> is one of the many options available eg -n to not have any progress indicator. <command> is the command to pass to lharc eg x to extract files. <archive> is the archive's filename.

To add files, or to create a new archive, the command "lha a" and then your file name will do the trick. To extract files, the command "lha x" and your archive file name will extract the archives file to

the current directory. To list what files etc are inside an archive, the command "lha v" will do just that.

NOTE: In some versions of LHarc the commands are caps sensitive.

There is also a variation of the program lharc called LHASFX, which you may come across.

Lhasfx is similar to lharc, but it adds a special header to the archive to allow the archive to be tested, listed and extracted by means of executing it. Neat huh? That means to us that you can create an lhasfx file and the end user doesn't need lha or lhasfx to extract it.

LhaSfx's command template is like this:

```
LhaSfx <Archive[.LZH or .LHA]>  
[<Destfile>]
```

Where <archive> is the name of the archive to convert. <destfile> is the name of the executable SFX-file to create. A '.run' suffix will be appended if it's not specified.

Basically, the average user of these programs (like me for example) will only rarely use other command or switches (there are myriads of them, I can tell you!). Without a doubt, the easiest way to get going with archives is via one of the excellent directory utilities that support these features. Take a look at Directory Opus for example, it has excellent handling for archive extraction etc. There are also workbench interfaces for these archive programs that can make life just a little easier, as you have gadgets and string requestors instead of all that CLI stuff. Also check out LHarc, which has full workbench style pulldowns and requestors built in.

There is also a program called LZ, and its command template is similar to Lharc's, and in many ways it is compatible with Lharc. Some people, like me for example, use LZ to unarc LhArc files, because it is often quicker than LhArc at doing so.

LZ [-options] <command> <archive> [file...] [destpath]

To add files the command "LZ a" followed by your archive's filename will do the trick.

To extract files "LZ e" or "LZ x" followed by the archive's filename will do just that.

To display the archive's contents "LZ l" or "LZ v" is what you need.

As you can see, very similar to Lharc in its usage but for a complete list of available commands and options, enter LZ via the CLI with no options or commands.

Next up we have UnArj, which allows us to extract ARJ type files. ARJ is a popular MSDOS archival program, so it is handy to have UnArj around.

The command template for UnArj is as follows:

UnArj <command> [-options] <archive>[.ARJ]

and the following commands are recognised:

- l** - to list the files in the archive.
- v** - for verbose listing of the files.
- e** - to extract the files.
- x** - extracts files and their full path.
- t** - to test the integrity of the archive.

The following options are recognised:

- c** - show archive comment.
- i** - to suppress the progress indicator.
- n** - extract only non-existent files
- q** - query on each file.
- x** - disable CRC checking (speeds up decompression by about 10%).
- y** - assume yes to all queries.

Then there is UnZip, which allows us to unzip .ZIP type files. ZIP, much like ARJ, is a popular MSDOS archive program, so again it is sometimes handy to have UnZip lying around.

UnZip's command template goes like this:

UnZip [-options[modifiers]] file[.zip] [filespec...]

The commands UnZip understands are:

- x** to extract files
- c** to extract files to the screen
- f** freshen the files in the archive without creating extra
- u** update the files, and create as necessary
- l** list files in short format
- v** list files verbosely
- p** extract to pipe, no messages
- A** test archive integrity
- z** display archive comment

Modifiers:

- n** never overwrite existing files
- o** overwrite files without prompting
- j** junk paths (don't make directories)
- q** quiet mode
- qq** quieter mode than quiet mode
- a** convert text (CR LF= LF)
- U** don't make names lower case
- V** retain VMS version numbers

That's about it for file archivers. Of course there are more around, especially in the MSDOS world, but for the most part, LZ and LhArc for the Amiga will keep you out of trouble, and having UnArj and UnZip in your C directory can save some stress when working with MSDOS based systems. Fortunately most of the archival programs for MSDOS machines have an equivalent program on the Amiga and they usually are compatible with one another.

Now for the whole disk archivers.

First up lets look at DMS, or the Disk Masher as it is sometimes known.

There are currently 2 ways of using DMS that I am aware of, via the CLI and there is a workbench driven interface called DMSWin which is offered to registered users of DMS who prefer a graphical interface.

DMS has these usages:

DMS Read File [.dms] [from Dev:] [TEXT filetext] [CMODE mode] [LOW lowtrack] [HIGH hightrack] [NOVAL] [NOZERO] [ENCRYPT password]

DMS Write File [.dms],,, [TO dev:] [LOW lowtrack] [HIGH hightrack] [NOVAL] [NOTEXT] [NOPAUSE] [DECRYPT password]

DMS Repack File [.dms] [TO file[.dms]] [LOW lowtrack] [HIGH hightrack] [CMODE mode]

DMS View File [.dms],,, [FULL]

DMS Text File [.dms],,,

DMS Test File [.dms],,,

DMS Help

Where:

Low = the lowest track of the disk to be processed, probably 0.

High = the highest track number to be processed, probably 80 for a full disk.

Cmode = compression mode from simple to heavy.

Noval = don't validate the disk on completion.

Encrypt = encrypt the file requiring a password to decrypt it

Nopause = don't pause to display banners (see text) or display bootblocks

Text = include a banner

There are more options and commands but, unfortunately I don't have any documentation about them. Don't worry though, the above commands are all those you need to mash and unmash a whole disk.

Next up is LHWarp.

LHWarp has the following command template:

LHWarp <read/write> <unit> <filename> <start track> <end track> <text file>

Where:

Unit = Drive number eg 0 for df0:, 1 for df1: and so on.

File name = Output or input filename.

Start track = Track number to start processing from, 0 to 79 (only valid in read mode)

End track = Track number to end processing, 0 to 79 (only valid in read mode)

Text file = Append text in <textfile> to output file, banners for example. (optional).

LHWarp is in many ways similar to DMS, and can generally be used in the same way.

Well that's about it, one point to note about these programs is that some of them are covered by copyright and some are also shareware. So if you use them a lot and like them, give the author a shareware donation etc and help promote these authors.

Before I leave you to your own devices, a quick few examples to help you along.

For example, we have just downloaded a ripper shareware game in LHarc format, ie the file ends with .lha and we want to give it a whirl.

Lets assume that we don't have Directory Opus or similar and we are going to drive via the CLI, *shock and horror!* Lets also assume that you have got lharc in your current path, I.E. you have it in the C:\ directory of the disk you booted from, or similar.

To help simplify matters lets make a directory on a disk to put this demo/game into. We will assume a blank floppy is in df0:, so if we open a CLI and type "makedir game", for example, to create a directory called game.

Now copy your lharc'd file into this directory and then simply type lha x demo.lha (or whatever the game/demo is called) and that's it. If you got it right you should see something happen in your CLI, like lha blah blah about the author then extracting file this and file that. Of course there are other ways to archive with the same results, this is but one of many.

LZ files can work the same way and in fact most lharc files may be extracted with LZ anyway. As an added bonus LZ is frequently faster than lharc.

Next issue, I hope to give a short review/tutorial on some terminal programs and what some of the jargon associated with these means.

Until next time, **BCNU!**

Oh, and by the way, keep sending in the mail, it's absolutely great to have feedback, be it positive or be it negative, more positive than negative though I hope! And don't hesitate to send in suggestions or ideas for future communications columns, if you need help, or are wondering about something, just ask, and I'll do my best to help out.

Regards,
Raff.

Raff Lerro

THE CRYPTIC CROSSWORD GAME

Vol. 24



Brian Gibson, Design and Compilation of Crosswords.



Andrew Pinnell, Programming.

WELCOME!
To Brian Gibson's
CRYPTIC CROSSWORD GAME

A Magical Mystery Tour

By Bill Wheeler

Why Magical, Mystery Tour? Magical because of our fascination with delving into new programming material, this was a new world for us all. Mystery, because of the subject matter and the convolutions through which our ideas were to take us and Tour, because, although we knew where we wanted to go - we didn't forsee all the backroads on the way.

I started off doing a crossword puzzle for our local user group (Tasmanian Commodore Users Association) and as every puzzler knows, new clues become harder to find as one month follows another. I became interested in producing puzzles within my puzzles as our club offered a monthly prize for the correct completion of the crossword.

I contacted Brian Gibson, a renowned Cryptic Crossword compiler for the "Australasian Post". Brian and I were to set up correspondence and develop a friendship with the aim of producing something to our mutual benefit, because Brian had been looking for someone to write his cryptic puzzles into software for the Commodore 64.

At this stage, pulldown menus and windows were starting to be it and this seemed to be the ideal medium to use as a format for Brian's puzzles. The question was, "what software was around, with which to write such a program along with doing the graphics for the puzzle?"

My partner, Barry Hill, and myself found a public domain product called "WEOS". Similar to the renowned product for the C64 "GEOS", WEOS was far better for our purposes. We had to include Brian's puzzle forms as he had written them, allow for a daunting number of clues and write them in a form that made them Fun, Fast and Factual.

Two years, as we learnt the programming skills and polished the concept into a finished product. Later we staged a demonstration at the user group and we knew we had a winner - but something was missing! Although everyone liked

using it, full colour, 36 pulldown menus and thousands of windows, unless they were cryptic enthusiasts they had difficulty in solving it.

Brian had written a booklet called "How to solve Cryptics" for his "Post" puzzlers which was given away as a prize for the best clue writing letter. So it was logical to put his effort to further use. Brian had written his booklet in a step by step manner, guiding his readers through the alleys of anagrams and hallways of hononyms - what they are and how they are used in cryptic clue writing - the mysteries revealed.

I set to work converting the booklet, chapter by chapter, into an interactive tutorial, again using WEOS, making it exiting as windows opened up, examples portrayed and set and exercises and answers explained - and another winner!

But why should C64 and C128 owners have all the fun? Enter IBM and Amiga...

I contacted Paul Nicholus, another Hobartian and C64 idealist seeking new ground in IBM. He could see the potential for IBM office workers, solving puzzles on the screen at lunchtime would be better than opening the daily paper to page 18.

Paul chose to re-program the C64 material using Turbo Pascal so it could be used under IBM's Windows 3. Paul also had to learn as he went along, certain aspects needed to be changed such as mouse operation and file manipulation - but enough of IBM.

Amiga - Andrew Pinnell, a fellow TCUA member, was starting to use AMOS and decided he would like to have a go at this project. Andrew also had much to learn about using AMOS and again converting the C64 material, but he did a great job in a short amount of time to produce another startling demonstration for the users group.

Andrew has also done a straight conversion of the "How to solve Cryptics" tutorial.

The C64, Amiga and IBM versions are written totally in Australia from go to whoa (or WOW!) and are now being marketed here and overseas.

The Cryptic Crossword Game **\$30.00**
How to Solve Cryptics **\$25.00**
For more information contact:

WSW Software
129 Chapel St
Glenorchy
Tas 7170



BRUSH MAPPING

using

IMAGINE

Part 1 By Mark Johnson

"Within Imagine" there are 4 types of brush mapping or wrapping if you like. The first and most common is colour mapping, the others are reflective, filter and altitude. Colour mapping is what it say's, the rest rely on gray scale images, I am going to focus on colour mapping in this issue. The most important thing first is, you need a colour image. It can be any size, in 2 colours to 16 million.

The first part of this tutorial we will make the objects we want to map. Instead of saying "hold the right Amiga key with ___" I will refer to it as "Amiga ___".

PICTURE

1. Load in a disk from the primary menu "F5", making it 200 in diameter and having 6 points. Pick the Disk, "F1" and select Pick Points "Amiga 3". Pick the top right point, and holding the shift key pick the right middle point. Join the points "Amiga J". Do the same for the left side of our object. Save the object as Picture.obj.

2. Pick the object "F1" and select Pick Point Mode "Amiga 3". Pick the centre point and delete it "Amiga D".

3. Select Pick Group "Amiga 1", and copy the object "Amiga C". Paste the object "Amiga P", pick it "F1". Enter the Transformation requester "T" and scale the X and Z axis 0.85.

4. Redraw the screen "Amiga R". The Z axis of the frame has more space than the X axis. We will have to Scale the Z axis only to match the X axis. Select Scale "s" and press "x" and "y", now scale the Z axis so it looks the same distance as the X.

5. Pick the two outlines and Select Skin from the Object Menu.

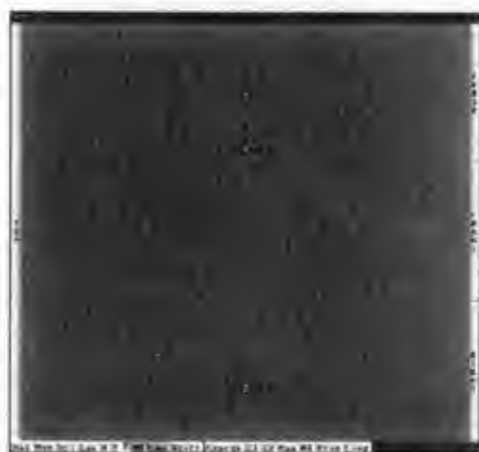
6. Select Mold "E" and click on Extrude. Type in 15 in the Length Box and Click Perform. Save it as Frame.Obj. Clear the screen.

CRAYON

Now for the crayon. We could use the sweep method on the crayon, it would be simpler, but we're going to use another method.

1. Load in a disk leaving the default settings. Pick the disk "F1" and select Pick Points "Amiga 3" pick the centre point and delete it "Amiga D". Select Pick Group "Amiga 1" and copy the disk "Amiga C".

2. Paste and select the disk "Amiga P" "F1" go into the transformation requester "Amiga T" and make the position of the Y axis=8. Repeat another 3 times for the following settings. Y axis=521, Y axis=525, Y axis=645. Redraw the screen "Amiga R" and zoom out a couple times. You should now have 5 disk's lined up in the Y axis. Fig



1
3. Pick the first disk, Select transform "Amiga T" Scale X and Z 0.950. Pick the fourth disk and scale X and Z 0.800. Pick the last disk and scale it 0.450. Click the mouse to deselect the disk.

4. Change the Pick Method to Drag box "F8" under the Mode Menu. Hold the shift key, drag and select all the disks. Select the "Skin" mode under the Object Menu, and wahlah, the outside of the crayon is done. Next is the ends to be filled in.

5. Select Pick Point "Amiga 3". Change the point mode to Hide Points, under the

Mode Menu. Using the drag box pick the first four disk's leaving the last disk. Select Add point "Amiga 6", change to Click mode "F6" under the Pick Menu. Zoom in twice "Amiga I". Using the front view add a point in the centre of the disk. Select Pick Point "Amiga 3", pick the centre point. Transform the point "Amiga T" and enter in the position X axis=0, Y axis=647, Z axis=0. Click Perform.

6. Select Add faces "Amiga 8". Starting with the centre point add faces to fill in the disk. Repeat the process above for the first disk, positioning the new point at X=0 Y=0 Z=0.

7. Pick the crayon "Amiga 1", enter the Transformation requester "T" and rotate -90 in the Z axis. Click perform. Click on Alignment "Amiga T" and type in Z=0, and click the "transform axis only" button. Click perform. The crayon should be lying down. Save as Crayon.Obj. Now clear the screen.

VASE

1. Add a axis "F4" pick it "F1" and select Add Lines "Amiga 9" from the Mode Menu. Make a half shape of a vase, FIG 2



2. Pick Group "Amiga 1" and select Mold from the Object Menu "Amiga E". Click on Sweep and leave the default settings. Save as Vase.obj. Now clear the screen "Amiga D".

Flat X and Flat Z

We are going to map the picture first. Load in the Picture.Obj you have just made. The picture axis is very important, standing the picture up in the front view and the axis being the same as the world axis (X axis being left to right, Y axis being front to back, Z axis being up and down) we're ready to begin.

Click on the picture axis, it should turn blue. Enter the attributes requester "F7", click on the "Brush 1" button and load in your picture. Leave Flat X and Flat Z as they are. "Edit" the brush. Scale the X and Z axis so the brush is just smaller than the picture.Obj.



Fig 3

Remember, the picture will only appear within the confines of the axis of the brush. If your brush is only on half the object, it will only appear on half the object. Render the object and if it look's OK, carry on.

Enter the Transformation requester "Amiga T" and scale the X and Z axis 0.95. Now load in the Frame.Obj, holding the Shift key pick it "F1". Group the two objects "Amiga G". Save the picture and clear the screen.

Wrap X and Flat Z

Load in the crayon. Note that the crayon axis is the same as the world axis. It is best to have the object's axis the same as the world axis, and the object facing the way that the brush should be wrapped in the Front view.

Enter the attributes requester "F7", click on "Brush 1" and load a Crayon.pic. (see main picture). Click on Wrap X and leave Flat Z as it is. Edit the Axis and scale the X axis so it fits on the main part of the tube. The Y and Z axis's have no purpose in this wrapping, so just leave them as they are. The crayon is done.

Flat X and Wrap Z

This is basically the same as the last one you have done. Load in the Vase.Obj,

pick it. "F7" for attributes, click on Wrap Z and leave Flat X as it is. You can resize the Z axis if you have a space at the top and bottom of the vase. Remember the Y and X axis don't do anything on this one, so don't touch them. It can cause funny things to happen to your wrap.

By all means EXPERIMENT!

Wrap X and Wrap Z

Load in a sphere from the primary menu

"F5", leave the default settings as they are. Enter the attributes requester "F7" and load in a marble.pic brush. Click on wrap Z and wrap X. Edit the brush to see what it looks like. That's it. Simple wasn't it!

The only thing I forgot to set was the colour attributes for the objects, you'll have to set your own. Now set up the stage editor and render away.

Signing off until next issue Mark Johnson.



Bit Master Software

PO Box 84, Tullamarine, Vic 3043. Tel (018) 538 225

Lotto Supersystems

Turn the odds your way and win!

Lotto Supersystems is the Smart, Cheap and Easy way to increase the odds of winning in lotto games, such as Pools, Tattsлото etc. Playing systems is the key in turning the odds your way and winning more often, however regular Lotery systems are far too expensive for the average person or syndicate to play consistently. Lotto Supersystems cuts the cost of regular systems entries, while maintaining the winning edge. This program will allow you to play much larger, more practical and affordable systems. Program includes, Computer System, Smarter number selection methods, User Freindly and completely mouse driven. Quick picks and much more.

\$25.00

PD Packs

GAME PACK 1 GALAXIANS - a great arcade remarkable. Plus Microbes (very fast) and Galaga. All on one self booting disk.	GAME PACK 2 SMURFHUNT - Hunt down the Smurfs and blow them away. Plus Venus Invaders and Tank Attack, a two player tank game.
PD Utilities Pack 1 This disk comes packed with some great utilities. Includes Dcopy, Zetovirus, DWedit, DWLasthope, Fixdisk and more!	Animation Pack This disk includes 3 entertaining animations by Eric Schwartz. * Vietnam Conflict * Shuttlecock * Stealthy Maneuvers 1MB required

All PD Packs only \$5.00ea. All prices include postage and packaging.

Send cheque or money order to:

BIT MASTER SOFTWARE
PO Box 84 Tullamarine Vic 3043

Name:
Address:

State

Post Code

Products Ordered

Total Enclosed \$

PUBLIC DOMAIN

=====

This month's selection of Public Domain programs comes courtesy of Gloria and Graeme Platt, system operators of Multiline 1990. Multiline is Western Australia's biggest Bulletin Board with thirty two lines, 3300 megabytes of on-line storage plus fourteen CD-Roms. If you have a modem call them on (09) 370-3333 and tell them you heard about it here.

SuperDark:

Some of you would have heard from your PC owning friends about AfterDark which is a selection of screen savers that are pretty good. The most famous being the 'Flying Toasters' which is a screen full of toasters with wings that do aerial stunts as they make their way across the screen.

Well now the Amiga has joined the party



with SuperDark. Not only can you select from Flying Toasters but you can have Fireworks and Fractals and a whole lot more. My favorite is the bouncing sentence where you type in a sentence and it bounces around your screen on a black background. Or maybe I prefer the one with the Tennis Ball musicians that play their instruments and bounce along to

a sound module you select. Anyway this great selection is up to version 1.5c now and is a Workbench 2 only commodity. Written by Thomas Landsburg, this is the thing to get.

Anyone for war? This game pits you



against your friends or the computer as you select from an arsenal of different explosives to launch at your opponents in an effort to blow them up. Don't forget your shields and maybe even some extra power. The computer opponents range from invincible to incredibly stupid and can offer a frustrating challenge. This is a version of an IBM game called Scorched Earth and while it has bugs it is still worth getting. Doesn't like Workbench 1.2 at all and I suggest you register it because the author needs our support to develop this game's potential.

Term:



This is one for those of you with accelerated machines and workbench 2. Although there is a version for 68000 machines it is way too slow to be any good when compared to some of the other terminal programs. This is a communications package that does everything except make coffee. Not only does it use external transfer libraries but it also uses external emulation libraries. Definitely one to get if your machine is a fast one and you use a modem. Latest version is 3.3 and it is freeware, although if you like, send the author something - even a postcard.

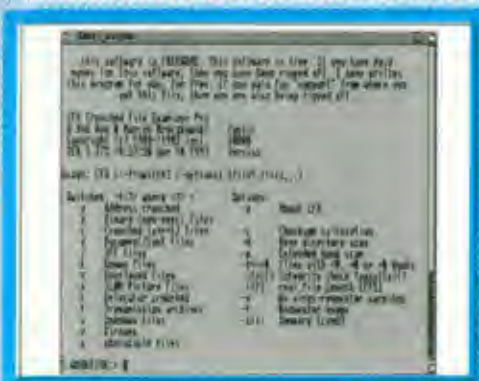
Programming Pals

It was suggested that a list of programmers be made for reference. This list would give the name and phone number of anyone who would make themselves available to answer "Help" type questions for new comers.

If you program in Assembly, C, Amos, CanDo, Basic or any other common language and you would like to participate, Please send your details to:

Programming Pals
PO Box 567
Mirrabooka
WA 6061

Crunched File Examiner (cfx) is great!
How could anyone ever live without it?



ZIP Compression method.

I just received an archive of the 'latest' Lha archive utility from an unknown source who uploaded it to my BBS. It is supposed to be version 1.48 but, funny enough, the only legitimate versions around are 1.38 for unregistered users and 1.50 for registered users. I didn't use this hacked Lha so I can't tell you what it does but it's a safe bet that if it's not real it'll do nasty things.

On the subject of Lha, for those of you thinking about registering I suggest you wait before you send your money off. After hearing from a friend that Stefan Boberg is not answering his E-mail I sent him some messages. It has been a month now and I haven't heard from him so all

Some people have pointed out that not everybody has a modem or knows where to get the Public Domain programs mentioned here. Well starting this issue I will publish the name and contact for some Public Domain libraries from around Australia. If you run a library or know of one not mentioned drop me a line and I'll include the name of the library next time. Also I would like to hear from people who have discovered or know of illegal version of programs so I can compile a list of questionable versions and keep people informed about them.

Dear Sir,

For some time now I have been trying to obtain some good shots taken from the Spaceshuttle or the moon. I even wrote to NASA but with no results. I am also in search of good colour graphics of military and civil aircraft.

Yours truly,
Don Dixon

All the best,
Bill

I hope this will encourage more people to write in and ask questions. If you can't find a picture you want, a utility you need or any other anything that might be on PD, just ask and we will attempt to answer.

New South Wales:

Prime Artifax - phone 008 252 879 (Sydney local call 879 7455). Big selection of compilations but a bit dear.

Victorias:

Bit Master Software

Phone (018) 538 225 see advert on page 27

Queensland:

Van Dieman Computing - compilers
of the OZAmiga coverdisk these guys
will be a real help. Phone : (075)
291117.

South Australia:

Leejan Enterprises - phone (08) 371 2655. Large selection and good value. Novices should try the Jumbo pack.

West Australia:

Amilight - phone (09) 367 4482. Huge selection of both normal and restricted (xxx) PD. Home of Twilight Zone Adult magazine.

I know I have missed some but these are only Public Domain libraries I have dealt with personally. If you know of any more write in and it will get a mention.

Don't Forget!

AMOS. If you haven't sent in your entries for the AMOS competition, then you had better be quick as final entries will be accepted no later than 30 July 1993.

Send entries to:

AMOS Competition
PO Box 567
Mirrabooka
WA 6061

Super Base

Superbase Professional has the reputation of being the best Amiga Database available. It features relational access, multiple indexes per file, programmable forms for on-screen invoices and the like, a programming language, an active data dictionary and an

easy to use, mouse operated environment. Superbase has been around since the days of the C64 and has been proven as a useful and worthwhile product. Recently, Superbase became the property of OXXI, who now market a Windows version of Superbase for the clone market. This change has resulted in Superbase 4 Professional being available for both the Amiga and IBM clones.

Superbase 4 Arrives

A friend of mine has been using Superbase since version 2. He upgraded to version 3, so it was only natural that he continue to version 4. His database earns it's keep looking after business records and hobby interests, so it was a good opportunity to test out this program in a real working environment. The upgrade from version 3 to 4 is not really an upgrade at all, you are really buying another Superbase. The price is around \$300 whether you owned a previous version of Superbase or not. So if you have already been following the upgrade path, prepare yourself for a few surprises.

Inside the box were two disks and three manuals. Unfortunately, one of the disks had almost self destructed in transit. The disks were just floating around inside the box. Not a good omen.

The biggest surprise was discovering that the Programming Guide was photocopied. Worse, it was badly photocopied. The diagrams were unreadable, the text on some pages disappears off the edge of the paper and pages were missing. The other two manuals were fine except that they also suffered from missing pages. The Applications Guide was missing an entire chapter.

The supplied example databases and programs were dated March 1991. The general appearance is of a pre-Workbench 2 product. All forms (screens) are in interlaced high res. An interesting paragraph from the Applications Guide says: "It is anticipated that users designing applications of the complexity of the Trading System will own systems capable of displaying in high resolution without flicker." I'm not exactly sure what that is supposed to mean.

After using Superbase for a while you begin to notice a few strange things. For instance, if you use Superbase 3 files the data occasionally disappears and reappears on the forms as you edit it. The manual states that Superbase 3 files, if reorganised with Superbase 4, can no longer be read by the older program. All files, forms and programs created with Superbase 4 should be regarded as incompatible with Superbase 3. This is because Superbase 4 is NOT derived from Superbase 3 Professional but it is actually related to Superbase 4 Windows for the IBM PC.

DML

Superbase has its own programming language so you can develop your own applications. The Database Management Language (DML) gives you full control of all of Superbases features and functions.

DML programs are entered using the supplied editor. I'm sorry to say that the DML editor is horrible. Menus are minimal, most functions are only available by key combinations, and those are not very obvious either. It seems to follow more of the IBM key strokes rather than the Amiga ones. The editor will

capitalise keywords as it scans each line, but is not very clever with syntax. That is, it will capitalise ANY keyword in a line, whether the syntax is correct or not. Errors are not reported until you actually try to run that line. This makes debugging a large program tricky.

The DML language is just like BASIC. Unfortunately I mean like AmigaBASIC (circa 1986) not like a modern basic such as AMOS. For instance, there are no procedures, you cannot pass parameters, all variables are global and there is

no way to draw any graphics except by loading in a Form. I wonder how you get Superbase to draw a graph?

Graphics

With Superbase 4, if graphics are not on a form, you can't have them. There are no drawing commands in the programming language, except by creating a form dynamically. Even then you are limited in what you can do. There are no diagonal lines or circles etc. One good feature is the ability to load an IFF picture and rescale it to fit a defined area. Superbase will also play sound samples and show external text files. All the above are accessed using a special type of field in your database that loads in the required file into a form as you scan the database.

Usability

Superbase applications will require a hard drive. Although a large program, it is also reasonably good with memory usage.

The supplied example programs work from a floppy, but are quite small. For real applications, floppy disk use is not recommended. Superbase Forms always load from disk. Expect your drive to get a work out if you try to design a large multi-form application. The famous VCR style control panel at the base of the screen allows quick browsing and searching of records. This is very easy to use and is a great help for the first time user.

There are quite a few good facilities available from the pull down menus. It is possible to print mailing labels for any file, create special purpose relational queries, update a set of files with programmable parameters, create views, transfer data via a modem, write text notes, import and export data in a variety of forms and perform a number of operations on files such as renaming etc. Other features can be programmed using the DML language. For the most part, Superbase performs its functions well.



The easy to use VCR

Form Designer

The Form Designer is basically good. The supplied tools and there are quite a few of them, work well. It takes a little getting used to, but once you get the hang of it the controls make sense. You can freely place fields, boxes, text and other objects about the screen. There is a grid feature, a ruler and other layout aids to help you design. There are some limitations though. You can't draw a circle or a polygon, or even a diagonal

line. Anything special and you will have to draw it using a paint program and import it. This makes any forms containing graphics quite big so they will take some time to load.

Superbase allows you to link different files on the one Form to create a multi-file application.

File Linking

Superbase allows you to link different files on the one Form to create a multi-file application. You can easily create an on-screen invoice, complete with transaction lines using data from different files. Superbase lets you link files using a special display that shows each file as a box on the screen. Simply select one file as the master, then select each file to link to it. The diagram shows how all the files are linked together.

The idea is excellent, but the implementation is so-so. You can't seem to link into the same file twice for example. When you do link a few files together, if the linking diagram extends off the screen, you can't scroll it to see the rest of the files. This type of problem should not appear in version 4 of a professional program.

File Handling

The main area that makes Superbase what it is, is its database filing system. Here Superbase earns some good points for the easy way you can define and use multiple files without worrying about how the whole thing works. When you define or change a database file, Superbase just does it. No need to alter your data, or recreate the files, it just happens. You can even modify a database file while you are using it. Very few programs have the same flexibility in this area as Superbase.

Superbase also features an active data dictionary. This means that the

Example Programs

Whoever wrote the example programs really likes the word GOTO. As a learning tool they are not very useful. There was no structure at all, GOTOs everywhere, GOSUBs were rare, the choice of labels and variable names almost meaningless and commands were not in functional blocks. The documentation for the examples seems to be written by same person. It says what each part supposed to do, but not HOW or WHY. I gave up trying to follow the GOTOs. If you know nothing about programming, DO NOT try to follow the examples, you will either end up confused or learn some very bad habits.

Programming Manual

The best way to approach the DML programming manual is "Don't believe everything you read." Be flexible when it says "See Also..." The keyword you are looking for may quite possibly be just an example under something else. Vast amounts are left unsaid about certain commands. You have to chase for HOW to use a particular command and quite possibly have to do other things in a certain order first. For example, you can REMOVE a file from the disk, but it doesn't say you have to OPEN it first. If you have lost the passwords to OPEN it, which is why you wanted to REMOVE it in the first place, you can't. But you can always DELETE it without needing the passwords, by using the DELETE command in a way the manual states doesn't work! In short, the manual is misleading and badly written.

Very few programs have the same flexibility in this area as Superbase.

Requesters and Things

Superbase has a nice Requester facility. There are dozens of predefined styles to choose from, each designed to cater for a particular job. You can easily bring up all sorts of lists, a file requester, various

information and query requesters, even a sorted selection of data from another file. Unfortunately you cannot create your own requesters. The selection is well thought out, so you won't have any problem finding one to suit your needs.

Why is everything in a window?

It would be nice to open a Form directly onto a screen without using a window, but someone decided the use Intuition windows for everything. Superbase follows the older style of Amiga programs,

rather than complying to Commodore's Style Guide.

Programmable menus are minimal as you cannot have sub items, but you can define key equivalents for common functions. Mouse control only knows about the left mouse button, there is no way to detect the right one. You can determine the mouse position on the screen, but as you can't seem to move or draw anything, it's not all that useful.

Overall

Here Superbase earns some good points for the easy way you can define and use multiple files without worrying about how the whole thing works.

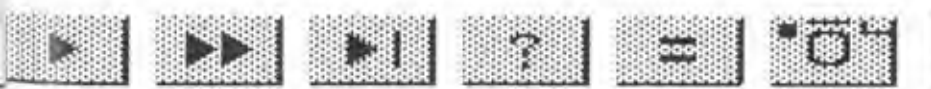
The core of Superbase is very good, the database system itself, but it's a bit like a lolly wrapped in horrible chocolate. There are a lot of stupid bugs and problems that just shouldn't be there. Some of them are just because of slackness, others you might forgive in a first release. But a program that has been around as long as Superbase, well, something is not quite right.

The program appears to suffer from the clone market mentality. Maybe we are supposed to buy a new version every 6 months just because they have fixed a few bugs and have "improved" it. I would prefer that the bugs were fixed before the program was released.

Superbase has had a good reputation for being the best database available for the Amiga, but if this version of Superbase is supposed to be the best around, then it's no wonder some people consider the Amiga a games machine.

I hope that OXXI can bring us the great software we need to do the Amiga justice. Hopefully there will be some sort of upgrade to address the problems of Superbase so it can regain its standing. If they follow the example of Soft Logic (Pagestream) or Europress (AMOS) by releasing patches or free upgrades to fix the bugs, then Superbase could again be the best available. But if they insist on charging \$300 a pop and not being too particular with their products, they may find sales a little slow in the Amiga community.

Neil McKnight



style control panel.

validation of data happens as you enter the data, not when you run a program. Each field can have its own validation and calculation formulas, even hookups into other files and help messages. As the data dictionary is active, if you change a validation formula or the definition of a field, it takes effect immediately. When an error occurs in a validation formula expect some weird error messages. Superbase knows there is something wrong, just not where.

David Perkovic does

Amiga Basic

Now start up AmigaBasic and type the following code into the window labelled "List".

```
PRINT "Please enter your name"
INPUT username$
PRINT "Hello "
PRINT username$
```

Hello and welcome to another Amiga Basic tutorial.

In my last column I introduced both the notion of a variable and the print and input statements. In today's column I will demonstrate how these two statements can be combined to form your own program. I will also introduce you to the if statement. Firstly let me give a very brief summary of my last column.

Variables

A variable is a labelled area within the computer memory where information is stored. A numeric variable can be labeled with any alphabetic character whereas a string variable can also consist of all alphabetic characters but must end with a dollar sign (\$).

eg.
stringvar\$ = "this is a string variable"
numericvar = 198

The above are an example of a declaration of a variable. When typed at a basic interpreter the variable stringvar\$ now has the value of "this is a string variable" and the variable numericvar has a value of 198.

Print Command

The print command allows the user to display text upon the monitor.

eg.
PRINT "Hi there"
prints "Hi there" on the screen.
PRINT a\$
prints out the value of the variable a\$.

Input Command

The input command accepts input from the user and places the input into a variable.

eg.
INPUT a\$
accepts a string from the user and places it in the variable a\$.
INPUT b
accepts a number from the user and places it in the variable b.

Once you have typed the above code in select the start command found under the run menu. The program will then run and ask you to enter your name, once you have typed your name in, the computer will respond with "Hello" followed by the name you entered. If the program doesn't run you most likely have entered the code in wrong so carefully check that the code you have typed in matches that above.

"Well", you may ask what use is the above program. It's main use is to explain the effect of input and output statements. The following program will do something useful by demonstrating how to add two numbers that you have entered.

DP Computing

Contract Programming
&
Computer Tuition
for the Amiga and IBM
range of computers

For more information contact:

DP Computing
11 Larrimah Rd
Morphett Vale
SA 5162

Phone: (08) 384 3352

```
PRINT "Enter a number"
INPUT number1
PRINT "Enter another number"
INPUT number2
PRINT number1
PRINT "+"
PRINT number2
PRINT "="
PRINT number1 + number2
```

The following is an example of what is produced when the above program is run:

```
Enter a number
? 1
Enter another number
? 2
```

```
1
+
2
=
3
```

As you can see the output 1 + 2 = 3 is written on separate lines this is due to the print statement automatically starting a new line once the given string, numeral or variable has been printed. To produce nicely formatted output we need to enter all the desired output with one print statement. This is done by replacing the last five lines of the above program with the following line:

```
PRINT number1 "+" number2 "="
number1 + number2
```

When the program is now run the output will be nicely formatted on one line.

Making Decisions

The ability to make decisions is a very powerful feature and basic allows decisions to be made through the use of the if statement. The syntax (i.e. the grammar rules of a language) of the if statement is as follows:

```
IF <expression> THEN
  <statement-block-1>
ELSE
  <statement-block-2>
ENDIF
```

An expression is a value that can either be true or false (i.e. "a=1" is an expression as it can be true if the variable a is 1 or false if the variable a is not 1). The first statement block is one or more statements that are executed if the expression is true while the second statement block again consists of a number of statements but they are only executed if the expression evaluates to false.

In simple terms, when the expression is true the first statement block is executed, ELSE execute the second statement block.

The following program is an example of how the if statement is used:

```
PRINT "Enter your age"
INPUT age
IF (age<50) THEN
  PRINT "You are young"
ELSE
  PRINT "You are getting old"
ENDIF
```

The above program, when run, will prompt you to enter your age, if the age you enter is less than 50 the expression "age<50" is true so "you are young" will be printed otherwise if the age you entered is 50 or greater "You are getting old" will be printed as the expression is false.

We can now use the if statement to write a program to add or subtract two numbers based on what the user has entered.

```
PRINT "Enter a number"
INPUT number1
PRINT "Enter another number"
INPUT number2
PRINT "Do you want to add or
subtract these two numbers (+ or -)?"
INPUT operation$
IF (operation$ = "+") THEN
    PRINT number1 "+" number2
    "=" number1 + number2
ELSE
    PRINT number1 "-" number2
    "=" number1 - number2
END IF
```

The above program will work as required if the user enters "+" or "-". An error arises when the user enters any other character or string, as the else statement will still be executed. To overcome this problem I will introduce the nested if statement. A nested if statement is an if statement within an if statement.

To give an example of this replace the if statement in the above program with the following:

```
IF (operation$ = "+") THEN
    PRINT number1 "+" number2
    "=" number1 + number2
ELSEIF (operation$ = "-") THEN
    PRINT number1 "-" number2
    "=" number1 - number2
ELSE
    PRINT "Error, please enter + or -"
END IF
```

The above statement checks to see if the operation is either a "+" or a "-" if it is then the correct operation is executed, if not an error will be printed.

Note the indentation of the above programs, this is to help the readability of the programs. I suggest you follow my method of indentation or create your own to make life easier when you start to write large programs.

To finish off I will leave you with an exercise to extend the above arithmetic program to include both multiplication and division. A full solution will be given in the next edition. Also in the next edition I will provide a brief summary of all the currently available basic compilers and interpreters.

Till then bye

David.



Well here we are for another edition of the Guild Hall. It's been a pretty slow two months for adventure games, at least for us Amiga users. The Ultima series has a new chapter for the IBM, making them four Ultima games in front of the Amiga. Maybe the rumour is true and no more Ultima games are being developed for the Amiga.

Eye of the Beholder III is out for the IBM, and the Amiga version shouldn't be far behind. If the IBM version is anything to go by this is going to be one hell of a game. It is bigger, meaner and nastier (if that is possible) than Eye of the Beholder I or II.

A lot of new releases are due out soon, hopefully very soon. Abandon Places II and Black Crypt II are due soon. The predecessors of these games were great and with luck the sequels will be better. There are a few new games being mentioned in overseas magazines but since I can't check out how real these games are I think I'll wait and see before I mention them.

Now for the surprise of the month, I found a Public Domain game that has been keeping me busy for days now. It's called HackLite and is a conversion of a game called Hack which is supposed to be well known in Universities. Set in an Ultima type scenario it is as involved as the Ultima games and yet I find it really addictive to play. If you would like to get your hands on it then run off down to your local Public Domain library and shell out the princely sum of five dollars for this game. Full documents and instructions are included and the range of character classes etcetera is full on.

Now this month I received a letter from Joseph El-Hayek in Guildford, New South Wales who is having trouble in Abandoned Places. He is stuck trying to find the first missing part of the sword. I didn't stick with this game so if anyone out there can help Joseph drop me a line here at OZAmiga and I'll pass it on. Joseph also wants to know if there is a number for Electronic Zoo, the makers of Abandoned Places, which he can call if he has any more trouble. I didn't have any luck finding one but if any one else out there knows it please let us both know.

I would also like to point out that I have enough solutions to Ultima V and Ultima VI so if you were going to send them to me please don't. On the other hand if you are stuck I can help you out.

Now for what you have all been waiting for, details of the Great GuildHall Competition. This competition is open to everybody, regardless of age. The judge - ME - has the final say and no one will change my mind without offering a huge amount of gold. Now the theme for this competition is fantasy. There are four ways to enter and you can enter as many times as you like.

Number one is artwork. Create your own fantasy masterpiece and send it in.

Number two is music. Compose a masterpiece of sound but remember the theme is fantasy.

Number three is text. Write a story. You can create your own world to set it in or use a game as the basis for your story. Minimum length is four pages and there is no maximum.

Number four is program. For those gifted enough to be able to program write a game, send it in. Or write a utility for existing games.

Anything to do with fantasy.

The winners will be judged on originality and effort. Anyone stupid enough to send in public domain artwork or program or to copy a story from somewhere will have his/her name entered in the GuildHall records as a DimWit and will be disqualified. The results will be published in the final edition of OZAmiga for the year and the winning artwork will be printed - in full colour - in the magazine. Winners from the other categories will have their work included on the coverdisk for all the world to see. So send your entries to OZAmiga, mark it as The Great GuildHall Competition and which category you are entering. Please include your name, age and address.

The Guild Master

A

SUPERFROG:

Here's some passwords for this game.

747822 - start at world one, level four.
446364 - start at world two, level two.

R

NOVA9:

The best concept I have seen for a while, this game is a must for accelerated machines and is still great on standard Amigas.

A

CTRL+ALT+HELP - repair ship and power shields.

CTRL+ALT+RETURN - gives ship lasers and rockets.

CTRL+ALT+UP ARROW - skips the level.

I

PREMIERE:

Great platform game. When it says 'Press fire to roll cameras' type SPARKPLUGS.

D

APIDYA

Type the following codes on the title screen followed by <RETURN>.

MISSHONEYBEE - to start the game on Level Two
HASTALAVISTA - to start the game on Level Three
DEPUTYOFLOVE - to start the game at level four
SNEAKPREVIEW - to start the game at level five
SHOWCREDITS - to see the end-of-game sequence

NOTE: it is possible that the cheats for level three and four are in the wrong order

BEVERLY HILLS COP

Click past the high-score screen and onto the select difficulty page. Now type MELLIE and you will be able to access all the games.

BLACK CRYPT

Play the game as you normally would (and collect as many items as possible). Save the game as Game A and without

moving, put all your items on the floor. Then save the game as Game B. Now quit the game and load a program like Diskmaster (just as an example) and look at the saved game files. Delete the character who is character B, change the name of character A to B and reload the game and play Game B.

CIVILIZATION

Press <SHIFT> and 1234567890t gives you a complete world map so you can see into enemy cities. This cheat appears not to have been included in newer versions of the game, but give it a try anyway! Could someone explain this cheat a bit better? It's a bit cryptic.

Press <ALT> and <R> to randomise the leaders' personalities.

DAYS OF THUNDER

Enter the qualifying race as normal and then press <P> to pause the game. Now type COMEFlyWITHME and the screen should flash the verify that the cheat is active. You can now pull <DOWN> on the joystick and fly into the air. Use <FIRE> to go forwards and the Function keys to get different views of yourself flying.

DEUTEROS

Go to the surface or the orbital stores room and press <SHIFT> and <C> simultaneously (the screen should turn green). Now press <SHIFT> and <C> again to resume. Now click on the master control icon and go to the stock screen. You will appear to have one of every item, when in fact you actually have an infinite supply of everything. Orbital space stations can now be built with rope frame section.

EPIC

Level 1 - AURIGA
Level 2 - CEPHEUS
Level 3 - APUS
Level 4 - MUS
Level 5 - PYXIS
Level 6 - CETUS
Level 7 - FORNAX
Level 8 - CAELUM
Level 9 - CORVUS

Try pressing <ENTER> a few times for a refuel, shield repair and a weapon boost up to 99 shots.

FIRE AND ICE

Press the <LEFT MOUSE BUTTON> on the title screen for infinite lives.

GODS

Select the 'enter password' option, type in SORCERY and you should now be able to start with infinite energy.

HARLEQUIN

Using the Space Hopper on water will allow you to skim across the surface without drowning.

Watch out for hidden bonuses located at the following locations:

- *The first tower's clock face
- *The clown's noses in the Learning Curve
- *The taps in the Sewerside
- *The trees and flowers in Cutesy Land
- *The pipe openings in the organ chamber
- *The ace of Hearts playing cards in the House of Cards

Don't stay in the Straws level too long. When the timer runs out, the credit tokens disappear and aren't regenerated. However, if you grab as many as you can and leave before the timer reaches zero, when you reenter the level, all the credit tokens should be back again. By doing this two or three times you will gain an extra life. When falling long distances, use the broolly power just before you hit the ground. It may only save you a little bit of energy, but every little bit counts. Heart pieces are located in the Rooftops, Hell, Cutesy Land and Sheet Muzak. Most levels change in some way during the game so always be sure there isn't an exit you've missed when you pass through a level which you have visited before.

Save your game frequently as it will take a long time to complete (roughly two and a half hours if you take the shortest route). Any levels which have water are bound to have a Fish power token nearby so don't venture near it without one. Something strange might happen should you hit the right notes in the Organ Chamber.

GOLDEN AXE

Play in One Player mode but with two joysticks. When you die press <FIRE> on the other joystick to get three extra lives.

FLASHBACK

EASY	NORMAL	HARD
BACK	PLAY	CLOP
LOUP	TOIT	CARA
CINE	ZAPP	CALE
GOOD	LYNX	FONT
SPIZ	SCSI	HASH
BIOS	GRAY	FIBO
HALL	PONT	TIPS

The action game of the 90's

SUPERFROG



Soon to arrive in this country is another of the Amiga's answers to Sonic the Hedgehog. Like Zool, this game is fast and furious, it will push your reflexes to the limit.

The object of the game is to find the keys to each level, thus progressing to rescue your girlfriend. You do this by collecting the number of coins shown on the large coin in the bottom right hand of the play screen.

After you have found the correct number of coins you must make your



©1993 TEAM 17 SOFTWARE LTD

LUCOZADE IS A TRADEMARK

FEATURES TABLE

INVISIBILITY	500	●
DESTRUCTO-SPUD	1000	■
RESTART COIN	500	①
SPEED UP PILL	500	⓪
SLOW DOWN PILL	500	Ⓢ
COIN	100	●
FROG WINGS	350	✈
EXTRA FROG	3000	★
BONUS FROG EVERY 200,000 POINTS		

DRINK LUCOZADE TO KEEP
YOUR ENERGY MAINTAINED

way to the exit. There are four levels on each world, each with it's own set of special difficulties. In the first world all is light and green as you make your way through the forest dodging bees, snails and birds.

The second world is within an ancient castle with many traps to watch for as you dodge slinkies, bats and fireballs. Each world gets progressively harder, putting more and more pressure on those

straining reflexes.

Set to be a big seller this game offers speed, playability, good graphics and very exiting sounds.

I thoroughly recomend this game to all arcade adicts.

David Reeves

P.S. Look in Arcadid for hints on this game



Lemmings 2 THE TRIBES

Those adorable little Lemmings are back in **LEMMINGS 2 - THE TRIBES**.

Psygnosis have really out done themselves this time. Usually sequels don't come up to scratch and with a theme like Lemmings DMA design had a hard job to do, but they've done it and done it brilliantly I must say.

This time the story is set on an island with 12 completely different and isolated Lemmings tribes. The island is in danger of destruction and the Lemmings must build an ark to take them to safety. The only way to power the ark is with the magic Talisman. Unfortunately the Talisman was broken into 12 many years ago and a piece given to each tribe for safe keeping. Now the tribes must make the journey to the center of the island where the ark is being built and reform the Talisman.

The 12 new Lemming tribes are; Cave Lems, Space Lems, Polar Lems, Highland Lems, Circus Lems, Classic Lems, Medieval Lems, Egyptian Lems, Outdoor Lems, Beach Lems, Athletic Lems and Shadow Lems where it's mission impossible time. Each set of levels for each tribe has it's own themed landscape features and theme tune.



Each tribe starts it's journey with an allotted number of tribelems. The aim is, as before, to guide them safely through the trials and tribulations of the current level and onto the next. Unlike the original Lemmings, there is no 'survival target' to aim for. As long as at least one makes it to the exit, then the level is complete. However on the next level you'll only start off with as many Lemmings as you saved which can make things tricky, even impossible.



The game starts out with a cute little animation of an old and wise Lemming telling the story of the island and it's trouble, to a young curly haired Lemming. The game then progresses to the option screen where you can play the game, pick which tribe you want to help, practice some of the new found Lemming abilities or load/save your current game.

As mentioned above the Lemmings have been given a host of new abilities that



include such bizarre actions as; pole-volting, missile launching, growing plants?, flying a hang-glider, skiing, swimming, diving, throwing snowballs? and turning in **SUPER-LEM!**. These are only a slice of the new things these little chaps can do. Also the levels have been given heaps more obstacles and objects to help you. Cannons are one of the more interesting ones, blasting your Lemming chap high into the air. Others include swinging ropes, trampolines, steam jets and bouncing balls?!

At the start of each level you are told which abilities you're allowed to use and like before you only have so many of each of them. After completing a level you are given either a gold, silver or bronze medal depending on how many lemmings you managed to get through. Each tribe must negotiate ten levels before reaching the arc, so 120 levels must be

completed to finished the game. Unlike the first Lemmings you can swap and change tribes all the time and go back to a completed level to try and do better. There isn't a password system but you can save your game onto a blank disk.

If you where a fan of the original Lemmings then I strongly suggest to look into this sequel, if you're not I suggest you look again as this game is brilliant. At times it will have you pulling you're hair out, and why not it can get hard at

times, but as you can change to another tribe at anytime it allows you a little breather to re-plot your plan. The graphics are great and atmospheric, the sound does the game justice and the playability? What can I say! It had me hooked for days. Can DMA Design outdo this one for Lemmings 3? I really don't know, but I can't wait to find out.

REVIEWED BY: CHRIS LEATHLEY

Dear Denise..



* Dear Denise,

I have recently shown my wife how to set up a data-base of her recipes on my AMIGA 2000 with 52 meg hard-drive; and I have created two drawers called "Savories" and "Sweets" within a drawer called "Recipes" on which to save her files. I have explained the process of saving a file many times but I still find her files scattered all over my hard-drive. She even managed to put a copy of the Data-base program itself in my "s" directory! Any ideas?

Exasperated
(N.S.W.)

Dear Exasperated,

I really think the time has come to take a firm grip on your wife's drawers and banish her files to a floppy disk. Add a line towards the end of your startup-sequence "lock <partition name> on" for each partition on your hard-drive; this will "write protect" your drive until you open a Shell and type "lock <partition name> off". When your wife boots up your AMIGA, she will only be able to save files to Ram, or a floppy disk.

Hope this is a recipe for success.

Denise.

* Dear Denise,

Quite a few friends and acquaintances have recently sold their AMIGA 500 and 2000 systems and now own IBM compatibles with "Windows", SVGA monitors, large hard-drives etc.

As the price of such systems continues to fall, Commodore seems content to sit on it's laurels, and the introduction of the AGA chip-set seems to be like shutting the stable door after the hens have roosted. The IBM compatible has become like a great octopus spreading its tentacles across the country, and I am really beginning to have my doubts.

Wavering
Yokine W.A.

Dear Wavering,

I think that should be "after the horses have roosted", and despite your mixed metaphors, your points are well made. A well equipped IBM can be bought very cheaply these days, but I think that you

underestimate the benefits which the AGA chips and other forthcoming innovations will bring to the AMIGA range.

Confidently (I think),
Denise.

* Dear Denise,

Why is the AMIGA Computer not advertised at all in the media here in Australia. The reaction of my colleagues at work when I inform them that I use an AMIGA at home is basically "What's an AMIGA?" They seem to think that it's some sort of Mexican cooking appliance.

Isolated,
VICTORIA.

Dear Isolated,

Believe it or not, Commodore Australia does have an advertising budget, but apparently they are convinced that half the money they spend on advertising is wasted. Unfortunately, they don't know which half! We can but hope for an improvement.

Ever the optimist,
Denise.

* Dear Denise,

I recently attempted to optimise the data on my 120 megabyte hard-drive as it was becoming rather fragmented. First of all, I backed up the drive onto a VHS tape using that new interface which turns your VCR into a tape-streamer; then got stuck into the drive with a public domain disk-optimisation programme. About ten minutes later, as I was making a cup of coffee, my dog accidentally snagged the extension cord which powers my AMIGA 3000, while chasing the cat which had been trying to eat the dog's dinner. The AMIGA of course refused to boot from the hard-drive and both my partitions now show up as NDOS.

I then discovered my wife in the process of taping an episode of "Burke's Back Yard" onto my VHS back-up tape! We got into a flaming row which ended up with her walking out on me (or driving rather, as she took my car). What do I do now?

Stuffed, Walpole W.A.

Dear Stuffed of Walpole,

Your "nom de plume" says it all really. You might consider this episode as Nature's way of telling you to give up computing; so after exacting suitable revenge on the dog and cat, I would suggest either killing yourself or selling your computer and taking up gardening as a hobby, which is much the same thing really when you think about it.

Yours in memorium,

Denise.

* Dear Denise,

My friends and I just love reading your amusing and off-beat column in every issue of OZAmiga magazine. However, reading between the lines, I think that beneath that flippant exterior, I can detect a warmhearted, intelligent, sensitive human being. Am I right?

Lesley Beahan, ACT.

Dear Lesley,

How astute of you my dear, Fate often casts us in a role which we just have to play out as best we can. In my assumed persona in this journal, I often feel like the young lady in the Victoria Park massage parlor who was asked to wear a clown suit for a bit of comic relief. It is only through what I feel is a gift in finding genuine pleasure in other people's misfortunes, that I manage to keep going.

Your caring, sharing

Denise.

That's all for now Amigoids, keep those letters coming.

Each edition the humorous backchat from Denise is brought to you by
Ian Harris from WA.

Portfolio

of an

Artist.

This issue's Artist is Shih Wei Wang of

Neutral Bay in NSW

If you would like to see your own work displayed here, send your images to:

OZAmiga Artists
PO Box 567
Mirrabooka
WA 6061



Shih uses an A4000
with an Opalvision
board linked with
programs like Imagine
and Real 3D

